EXHIBIT L

FOR OFFICIAL USE ONLY

SECURITY PROOF OF CONCEPT KEYSTONE

NETRANGER REAL-TIME NETWORK INTRUSION DETECTION
Performance and Security Test

Prepared for:
Maryland Procurement Office
9800 Savage Road
Fort George G. Meade, Md. 20755

Contract:
MDA904-96-C-0215

By
COACT, Inc.
9140 Guilford Road, Suite L
Columbia, Maryland 21046

Document No. 010511

30 April 1997

5/5/97

Not releasable to the Defense Technical Information Center Per DoD Instruction 3200.12 Distribution limited to U. S Government Agencies Only. This document contains NSA Information (Jan.1992). Request for this document must be referred to the Director NSA.

FOR OFFICIAL USE ONLY

Memorandum

To:	SPOCK Consortium
CC:	
From:	Bill Marshall
Date:	May 1, 1997
Subject:	SPOCK Demonstration Report - NetRanger Security Architecture

The SPOCK Consortium, as a part of its continuing goal to explore INFOSEC commercial solutions and enabling technologies, is pleased to issue this demonstration report on the Wheel Group NetRanger Real-Time Network Intrusion Detection Product.

This report demonstrates V2's cooperation with industry to jointly participate and validate security features of commercially developed products. This report validates all of the vendor security claims. It also documents the results of the demonstration of the Wheel Group NetRanger Real-Time Network Intrusion Detection product in a realistic warfighter application The NetRanger report provides an overview of the security functionality and significant lower level information to the government accreditation authorities, system integrators, and security architects which should be highly beneficial in determining their security needs. The vendor may find this report of value in charting the product's evolution.

Bill Marshall
Chief V2, NSA SPOCK Chairman

Table of Contents

TRADEMARKS

Data Privacy Facility	NSG (Storage Tek Network Systems Group)
BorderGuard 1000	
BorderGuard 2000	
Network Systems	
Passport	Nortel (Northern Telecom)

Document Introduction

This document is a SPOCK (Security Proof of Concept Keystone) report on the WheelGroup NetRanger Real-Time Network Intrusion Detection Product.

NOTE: Appendices may not be included with the document. They are quite large, and contain massive support data. Individual Appendices can be requested through appropriate channels.

Content. This document consists of the chapters shown below.

Abbreviations and Acronyms which may be Used In This Document

Storagetek Network Systems Group (NSG)	1
Data Encryption Standard (DES)	1
International Data Encryption Algorithm (IDEA)	1
Network Systems Cipher One (NSC1)	1
Network Sensor (NSX)	1
wide area network (WAN)	2
Air Force Information Warfare Center(AFWIC)	2
Fleet Information Warfare Center (FIWC)	2
Land Information Warfare Activity (LIWA)	2
Security Proof of Concept Keystone forum (SPOCK)	5
National Security Agency (NSA)	5
Information Security (INFOSEC)	5
Army Battle Command Battle Laboratory (BCBL)	5
Internet Protocol (IP)	10
Internet Control Message Protocol (ICMP)	14
Security Analysis Package Daemon (SAPD)	22
Virtual Private Network (VPN)	25
Fiber Distributed Data Interface (FDDI)	25
Integrated Services Digital Network (ISDN)	26

Figures. Tables. and Graphs

Chapter 1

1. Executive Summary

Validation and Performance testing was performed on the WheelGroup Corporation NetRanger Real-Time Network Intrusion Detection system that includes Storagetek Network Systems Group (NSG) BorderGuard 1000™ and 2000™. *NetSentry* and *Data Privacy Facility* (*DPF*) modules (which provide Network Administrators the ability to control and encrypt packet traffic through the NSG security devices) were an integral part of the software tested. Encryption algorithms included in the DPF package are Data Encryption Standard (DES), Triple DES, International Data Encryption Algorithm (IDEA), and Network Systems Cipher One (NSC1). Only IDEA was used for this series of tests. Objectives of the tests were to independently validate WheelGroup claims concerning the performance and technical security capability of the system's hardware and software package. Claims to be verified were:

When the Real-Time Network Intrusion Detection system is installed and configured properly (see attached network configurations) the package will provide practical and effective:

1). network attack detection,

2). director to director, and director to Network Sensor (NSX) data privacy (confidentiality) across unsecured networks,

3). auditing and reporting of detected attacks,

4). remote configuring of an implemented 'security policy' as reflected by the suite of 'content' and 'context' filters operationally installed,

5). selected protection by enabling automatic responses to perceived attacks or allowing manual intervention to protect circuits/data streams.

Performance testing and the exercising of security functionality testing were performed over a seven site network. An Internet based wide area network (WAN) was used for interoperability and penetration testing. The WAN was comprised of Internet connections between the following sites:

1) Army Battle Command Battle Laboratory (BCBL) at Fort Gordon, Georgia
2) NSA/V2 at Fort Meade, Maryland
3) Air Force Information Warfare Center(AFWIC), in San Antonio Texas
4) Center for Integrated Intelligence Systems (Space and Naval Warfare Systems Command) in McLean, Virginia
5) Fleet Information Warfare Center (FIWC), Norfolk, Virginia
6) Land Information Warfare Activity (LIWA), Fort Belvoir Virginia

COACT Inc. (NSA) in Columbia Maryland was used to monitor activity at all sites and act as an initiation point to launch scripts testing the capabilities of the product as implemented for this test.

Two series of tests were conducted to validate the above listed claims. Results of the tests clearly demonstrated that when properly configured, the NetRanger hardware/software package:

1) Can be used to detect, report, and act on intrusion related activities launched across a network with a high degree of accuracy,
2) Would detect all attempted penetration signatures contained in the default list as installed in the NetRanger for this demonstration,
3) Can be used to provide practical and effective intrusion detection, reporting, and selected automatic response actions,
4) The router/software combination will pass traffic transparently at 100Mbps or faster rates,

5) The Directors and NSX's can communicate securely using the encryption support provided in the NSG BorderGuard Security Devices.

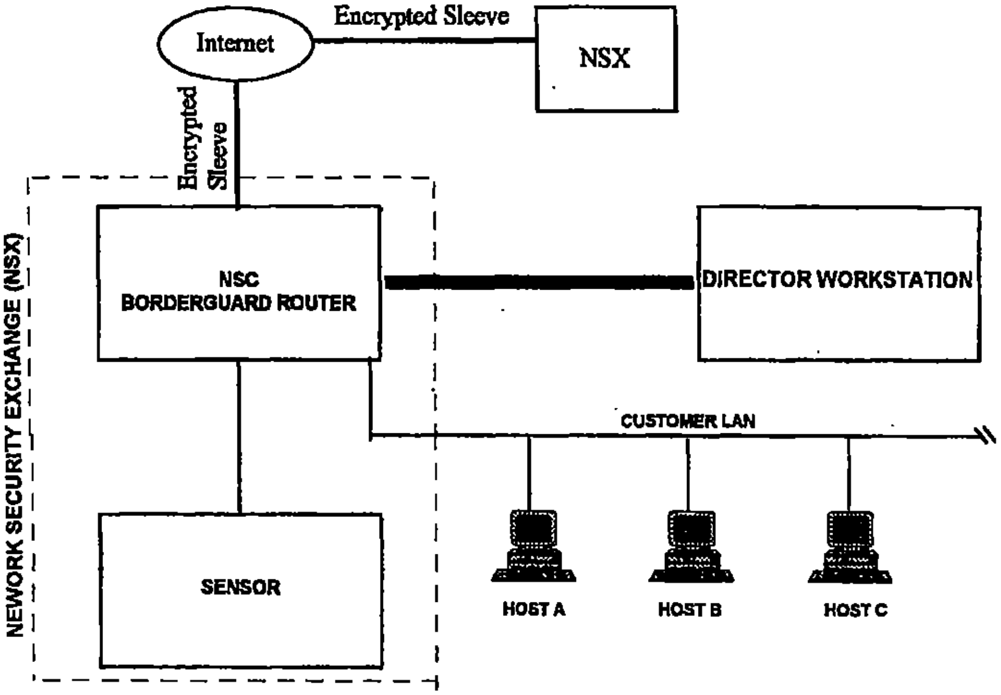

Figure 1-1 A Typical Site Configuration

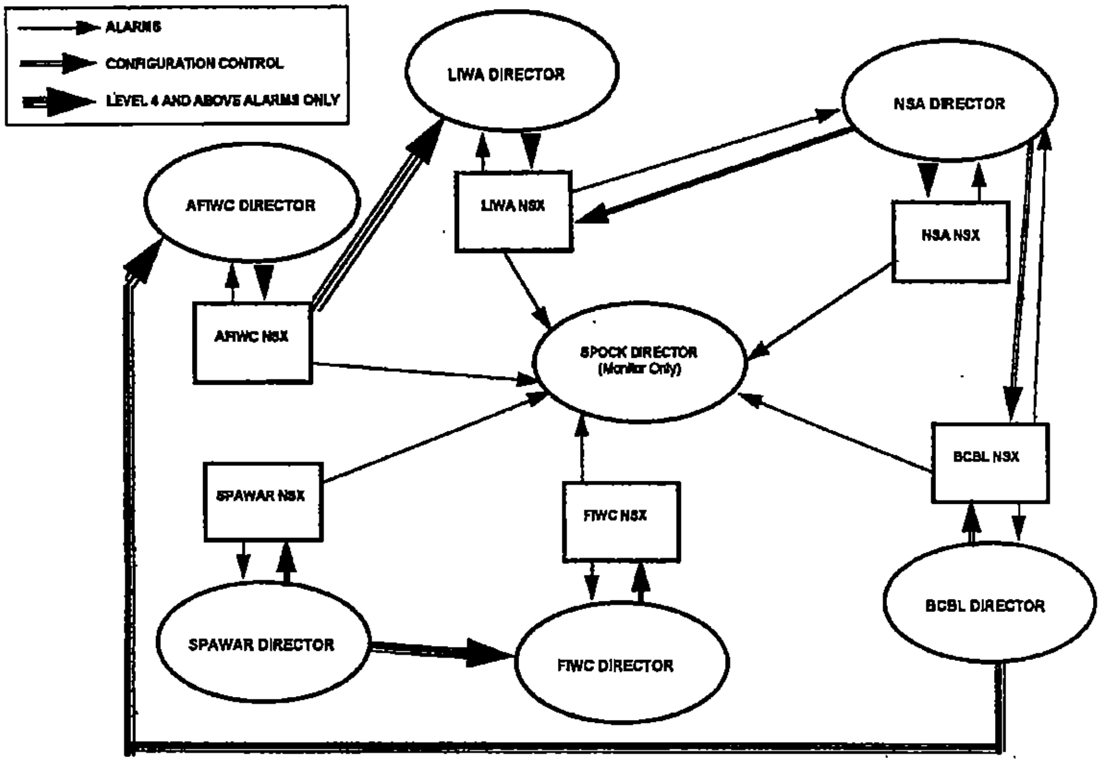

Figure 1-2 NetRanger Connectivity/Hierarchy

Chapter 2

2. Introduction

The Security Proof of Concept Keystone forum (SPOCK) is an National Security Agency (NSA) sponsored joint government-industry consortium, interested in exploring Information Security (INFOSEC) commercial solutions and enabling technologies. As a part of the charter, SPOCK participants conduct proof-of-concepts to demonstrate security features of commercial and government systems that can support dependable security architectures. Testing outlined in this document was conducted to validate WheelGroup Corporation claims of performance, technical security, functionality, and interoperability of the Real-Time Network Intrusion Detection system in an operational network architecture. Performance testing and the exercising of security functions tests were performed over a seven site network. An Internet based virtual wide area network (WAN) was used for interoperability and penetration testing. The WAN was comprised of Internet connections between the Army Battle Command Battle Laboratory (BCBL) at Fort Gordon, Georgia; NSA/V2 at Fort Meade, Maryland, Air Force Information Warfare Center in San Antonio Texas, Center for Integrated Intelligence Systems (Space and Naval Warfare Systems Command) in McLean Virginia, Fleet Information Warfare Center, Norfolk Virginia, and Land Information Warfare Activity, Fort Belvoir Virginia. COACT Inc. (NSA) in Columbia Maryland was used to monitor activity at all sites and act as an initiation point to launch scripts testing the capabilities of the product as implemented for this test. Performance testing was conducted by government and contractor personnel and monitored by other SPOCK representatives at the seven sites. Penetration testing was performed by BTG contractor personnel using scripts fully reviewed and approved by the SPOCK participants.

The WheelGroup NetRanger is a commercial system that provides Real-Time Network Intrusion Detection, reporting, and dynamic selective intervention. NetRanger is composed of sensors, called NSXs, and Directors. These components report alarms and dynamically re-configure security policy using secure encryption provided by the NSG BorderGuard Security Devices.

The NSG BorderGuards encrypt data using standards-based cryptography, including DES, Triple DES, IDEA, and NSG's private encryption algorithm, NSC1. Only IDEA was used during this series of tests. The evaluated NetRanger hardware used includes The WheelGroup NSX sensor, the BorderGuard 1000 and 2000, and the Passport Security Devices. The NSX sensors are used to determine violations to the security policy. A Sun Sparc 5 was used to launch the scripts and Sparc Ultra 1s served as hosts for the Director functions.

2.1 Claims to be Verified

The WheelGroup Corporation states that their NetRanger Real-Time Network Intrusion Detection package will :

1) report and selectively protect against network attacks, based on the security policy selected by the user and implemented within NetRanger,
2) be effectively transparent to the data stream at 100 base T rates,
3) provide data privacy (confidentiality) between the Directors and NSX sensors across unsecured networks.

Testing was conducted to determine the accuracy of the following statements in support of the above listed claims:

A. The NetRanger Sensor (NSX) can:

1) be deployed in bridge or router configuration,
2) be used with network throughputs from 56Kbs up to 100Mbps,

3) perform real-time intrusion detection, notification, and response,
4) configure event *severity* levels,
5) audit system activities,
6) configure command authorization,
7) automatically provide log file management.

B. The NetRanger/NetSentry/DPF combination can be configured to provide practical and effective data privacy Communications Architecture between a Director and Network Sensor (NSX) across unsecured networks by:

1) guaranteeing all NSX/Director communication,

2) securing all NSX/Director communication,

3) allowing flexibility in NSX/Director communication through configurability options.

C. The NetRanger Director can:

1) provide centralized command and control of multiple NSX sensors,

2) provide displays which are configurable and flexible,

3) notify off-duty personnel of events,

4) provide stage data to a relational database for subsequent analysis.

2.2 Security Goal

Verify the claims made in Paragraph 2 for network attack detection, data privacy (confidentiality), and event notification with selective invoking of port protective measures.

2.3 Functional and Performance Goals

1) Measure performance *impact* of the NetRanger on *throughput of monitored data* passing through its detection filtering process.

2) Measure performance impact of encryption on data throughput.

3) Test functionality of intrusion detection filters.

Chapter 3

3. Penetration Testing

3.1 Introduction

NetSentry has considerable means at its disposal to provide penetration protection. This functionality is provided through its audit generation and dynamic shunning capabilities. This report presents the tests, rationale, and results of each exercise directly applicable to penetration protection or intrusion detection.

3.2 Phase 1 Tests

The first series of penetration tests completed were conducted in a sequence/response environment, whereby an non-intrusive attack was launched from a central location to selected sites, the response (i.e. alarm generation/shun action) noted, and the sites then re-configured or altered prior to the next sequence. While it was possible to launch the non-intrusive attacks to all sites simultaneously, the participants agreed that two sites, previously chosen by the SPOCK committee at random, and varying with each event, would be sufficient to exercise the product and verify each claim. This phase required reconfiguring or resetting the site software after each event, and these arrangements significantly minimized excessive the coordination between COACT (the launch site) and the target sites.

3.2.1 Configuration and Equipment

An overview of the test network is shown in Figure 1-1. The equipment used included NSG's BG1000™ and BG2000™, and Passport™ Security Devices, Sun Sparc 5 and 20 Workstations, and appropriate software (HP Openview, etc.) The platforms were configured to connect to the Internet, from where attacks are likely to be launched. Multiple sites were linked with each other via the Internet and protected against network intrusion by the NetRanger System.

3.2.2 Tests and Results

All testing was conducted from the site at COACT, and non-intrusive attacks were launched through an ISDN connection to the Internet. All non-intrusive attacks were directed against the six participating sites to specific IP addresses as appropriate. These Internet Protocol (IP) addresses were agreed upon and verified during the SPOCK script writing sessions with the participants. The NSX's recognition of reception of the attacks, and their output to the respective site Directors (based on previously set alarm thresholds) were also monitored at COACT over the Internet on a SPARC ULTRA 1, configured as a Director. Tests were run based on a detailed test plan. (see NetRanger Capabilities and Test Scenarios, Version 1.2, dated 24 February 1997, contained in Appendix A.) Each test is referenced to the appropriate paragraph in that document.

Test 1: (see Append A, para 1.4). Verify that the NSX sensor can be configured to send user-defined alarm levels. Run a non-intrusive SATAN attack and verify a level 5 alarm is generated. Then change to level 4 and repeat the process. Note that levels set below the alarm threshold will not be displayed on the Director, but logged into the NSX audit file for subsequent downloading.

Results: The same input (frequency of event) resulted in a display on the Director when the frequency met the alarm level qualification, and subsequent non-display when the alarm level was set to a higher level.

Test 2: (see Append A, para 1.5) Verify that the NSX sensor maintains a log of all errors and commands. Ensure at least one destination in an NSX's destinations file is configured for level 1 ERRORS and COMMANDS. Issue a series of commands from the Director and verify that they are logged into the appropriate destination's log file. Then, misconfigure one of the applications on the NSX to generate an error and verify it is logged.

Results: When a wrong password was sent to the BorderGuard™, or Passport™ the NetRanger logged the event.

Test 3: (see Append A, para 2.1.1) Verify that the NSX sensor queues information when *connectivity* to the Director is lost and when the link is re-established that no *information* is lost. Disrupt the connectivity between an operating NSX and Director. Cause alarm to be generated that would normally be sent to the Director (i.e. reportable alarm level). Re-establish connectivity and verify queued messages are re-sent.

Results: A port sweep strobe non-intrusive attack was sent to the disrupted targets. When reconnected, level 4 and 5 reportable alarms were retransmitted as claimed.

Test 4: (see Append A, para 3.1.3) Verify that an operator can manually control an NSX system's responses to events from a Director system. Highlight an alarm associated with one of the NSX systems displayed on a Director Security map. Apply and remove the shun for this attack.

Results: The site operators successfully enabled and disabled the shun feature. The launched non-intrusive attacks were successfully repelled.

Test 5: (see Append A, para 3.2.2) Verify that multiple NSX systems can transmit different alarm thresholds. Force three different NSX systems to generate level 2, 3, and 5 alarms in order. Verify that the default Director setting produce green (normal), yellow (alert) and red (major attack) icons. Reset the alarm thresholds to their defaults.

Reset one of the NSX system thresholds to level 2 alarm and another NSX system to level 5 alarm. Run the same attack and verify all icons are yellow.

Results: UDP Port Sweep non-intrusive attack generated level 5 alarms. Mail recon probe generated level 3 alarms. The icons generated represented the appropriate color for the level set in the alarm sensitivity table. The exercise verified that the boundaries for icon generation can be set and the alarms will register within the set boundaries.

Test 6: (see Append A, para 3.2.3) Verify that a Director can consolidate duplicate alarms events into a single icon. Set default consolidation threshold for duplicate alarms to 2. Launch an attack for which a signature is resident, three or four times. After three successive attacks, the Director will display a single ICON with a counter indicating how many instance of that attack has been detected.

Results: Port sweep (Strobe) non-intrusive attack launched three times. The fourth consolidated on the target Director from three icons to one as claimed.

Test 7: (see Append A, para 3.2.4) Verify that a Director can propagate alarms up through an icon hierarchy. Generate an alarm icon of marginal status and verify that it propagates up through the Director display hierarchy (i.e. NSX application, NSX system, NSX collection, to root NetRanger). Reset the display and repeat with an alarm of critical status.

Results: Level 2 non-intrusive attack signatures for the TCP port sweep were previously validated. A level 5 strobe non-intrusive attack was subsequently conducted and icons registered on the monitor as yellow, multiplied, and then a red icon appeared with them, validating the migration of the alarm through the NetRanger hierarchy.

3.2.3 Conclusions

All phase 1 tests, requiring close coordination between the network monitoring site (COACT) and the six participating sites were satisfactorily accomplished with a minimum of problems.

3.3 Phase 2 Tests

This second series of penetration testing was conducted over the same networks used for the phase 1 tests. The difference was this series of tests did not require the sites to accomplish any reconfiguring or resetting of the Directors as a result of the non-intrusive attacks. (i.e. tests not sequenced in order with corresponding actions at the distant ends.) An overview of the network is shown in Figure 1-2.

3.3.1 Configuration and Equipment

The equipment and software configurations remained as during phase 1 testing.

3.3.2 Tests and Results

Test 1: (see Append A, para 1.3.1.1) Verify that an NSX sensor can detect a *context*-based attack and generate an alarm either/when (1) an attempt is made to access an 'unauthorized service' on a host protected by an NSX system, or (2) an 'unauthorized external system' attempts to access the protected network. Ensure 'rlogin' is disabled in the default configuration. Attempt rlogin onto the protected host. Secondly, permanently block an external host by creating a 'last block fail' filter including the host's IP address. Transport the filter on to the BorderGuard™, or Passport™ and compile it. Apply it to the last filter point. Then, using Telnet, try to reach the protected host from an *unauthorized* host. An Internet Control Message Protocol (ICMP) 'ICMP unreachable' alarm should appear on the Director Security Map.

Results: COACT tried to telnet (as an unauthorized host) to the protected host sites. An 'ICMP Unreachable' alarm appeared on the sites Directors verifying the claim.

Test 2: (see Append A, para 1.3.1.2) Verify Net Ranger detects misuse based on the content of network packets. Verify that the Network sensor can by default, detect and produce alarms for many standard user attacks and user-defined content-based signatures. Execute sendmail attack and observe alarm generation.

Results: The sendmail non-intrusive attack was executed using both standard content and spurious information (i.e. upper and lower case characters, continuous spaces, etc.) Specified keywords, for which signatures had been generated, were included in the mail. The system recognized the attacks and generated the requisite alarms.

3.3.3 Conclusions

The suite of features evidenced in the NetRanger product verified herein contribute greatly to the prompt detection of unauthorized attempts to penetrate a protected environment, and monitor authorized events for adherence to proper policy.

Chapter 4

4. Access Control Functions

4.1 Introduction

The purpose of this testing was to verify the WheelGroup Corporation Security claims related to Access Control. Testing involved the verification of the security of the NSX sensor/Director communication path and access control features built into the product architecture.

Each trusted enclave consisted of a Director/NSX sensor combination with a encrypted sleeved connection between them. Additionally, within the seven site network, six site NSX sensors were connected to the seventh site's Director (monitor) via encrypted sleeves.

For the majority of this testing, the non-intrusive attack launch computer (laptop), located at the seventh site, did not use an encrypted sleeve. (It was only used during pre-testing checkout of the network.) Various access control features were verified using this launch terminal, addressing commands to the host IP addresses at the six other sites as the test scenarios dictated.

4.1.1 Configuration and Equipment

Figure 1-2 shows an overview of the test configuration. The test configuration contained six Sun and one HP workstations configured with HP OpenView software and one Gateway 2000 laptop configured with appropriate files for launching non intrusive attacks per the agreed scripts.

Testing and Results

The following tests were performed:

Test 1: (see Append A, para 1.6) Verify that access to an NSX sensor is dictated by its access control list. Confirm that an action such as 'getbulk' is authorized by the Director's authorization table. Execute 'nrConfigure' command and verify. Delete 'getbulk' from the Director's authorization table. The command should fail the second time.

Results: The 'nrConfigure' command was accepted, and after removing 'getbulk', the command was subsequently rejected.

4.1.2 Conclusion

The access control feature performed as claimed. When understood and maintained properly, this feature can contribute to the overall security of the NetRanger functionality.

Chapter 5

5. Performance Testing

5.1 Introduction

This chapter describes the performance tests conducted on the NetRanger/NetSentry combination product. These tests were performed over a seven site network created for the purpose of this exercise. Five sites connected their NSX sensors to their Directors via their local ethernets, and WAN connectivity was accomplished using T1 links. The sixth site (BCBL) was limited by virtue of a 256 kbyte pipe, shared with all other users located at Ft. Gordon. The seventh site was connected to the other six via a 64 kbyte ISDN link to the Internet. (The actual network bandwidth was unknown because many of the paths were undoubtly shared by other users and varied during the tests.)

The primary goal of the performance testing was to verify the features of NetRanger which did not directly relate to access control, intrusion detection (penetration related), or encryption. A claim was made that NetRanger could support up to 100 megabit data thruput without any significant impact on the traffic itself. *A T3 connection could not be located for this test so that claim was not addressed by SPOCK. (see para. 5.3.1, Test 2, this chapter.)*

5.2 Test Device, Configurations, and Tests

The orchestrated scripts required no *special* test devices. The configurations were the same as used for penetration and access control.

5.2.1 Test Device

One of the six sites, (NSA), possessed a LAN Analyzer, which was used to verify the traffic flow as to when it began and stopped.

5.2.2 Configurations Tested

Figure 1-2 shows the testbed configuration.

5.3 Tests Performed

Test 1: (see Append A, para 2.1.2) Verify that the NSX sensor will send information to the Director via alternate routes as links are broken and re-established (i.e. fault tolerant). Identify a primary route to one Director and a secondary route to the same Director within the test network. Bring the NSX on line to verify that alarms are being received at the Director. Execute a 'nrconns' command to verify the primary route. Break the connection for the primary route and send alarms. Verify no alarms are lost. Verify the reason for continuity by observing NetRanger Director reports showing secondary access path established.

Results: Two logical paths were established vs. two physical paths. (allowable within the vendor claims) through configuring a 'dual-homed' Director. The NetRanger screen reports and the alarm audits conducted at the site verified secondary connectivity with no lost data.

Test 2: (see Append A, para 3.1.2) Verify that a Director can manage Multiple NSX Directors. Perform several 'get' and 'getbulk' commands against one or more NSX's. Then, change the default alarm configurations, ex. DEFCON1 to DEFCON2. Provide auto-shunning in the second configuration.

Results: Two sites had their configurations changed by a third site.

Test 3: (see Append A, para 3.3) Verify that the Director can send pagers and e-mail in response to an event. Configure a Director to generate page notifications or distribute e-mail based on thresholds defined in the NetRanger software. Then verify that e-mail and/or pager notifications are generated in response to test attacks. Compare the output of these notifications with information associated with the corresponding Director alarm notifications.

Results: A site with alpha-numeric paging capability was successfully notified of the attack as claimed.

Test 4: (see Append A, para 1.7) Verify that an NSX sensor automatically transfers Event and IP Session logs to an archive device. Establish proper file management thresholds for the NSX's sapd. Execute an 'nrget' command against the sapd token FileMgmt. The returned information will include directory size thresholds for archiving Event and IP Session log files. Assure the threshold is set to a lower value than the size of the logs accumulated during the test. Then configure sapd's Dump_OFFLINE feature. This instructs the sapd to execute the appropriate script, dumping the logs whenever the threshold is exceeded. Then define the commands to transfer to the appropriate archive device. Commit the new configuration to disk and verify that events are being logged. Finally, verify data has been successfully transmitted.

Results: The automatic backup of the event and IP logs, in parallel with normal operation of the NSX sensors was verified as claimed.

Test 5: (see Append A, para. 3.2.1) Verify that the Director provides a description of each security event. Highlight an alarm associated with one of the NSX systems displayed on a Director security map. Select the appropriate command from the HP Openview menu and access the 'attributes' dialog. The system will display the time of the event, event type, alarm level, source address, destination address, and so forth. Verify that the information varies depending on the event .

Results: The attributes dialog was found to contain the information as stated.

Test 6: (see Append A, para 3.4) Verify that a Director can stage event data to a relational database. Also verify a Director detects data staging errors and rolls back incomplete data transfers. Install Oracle and configure access to it. Configure Security Analysis Package Daemon (SAPD). Verify the logs are being created and queued for database transfer. Start SAPD process. Generate log events by initiating several attacks against hosts protected by one of the NSX systems. Verify the SAPD run history. Verify the data has been successfully loaded into the database by running one or more queries. Verify the database has been properly loaded. Disrupt database connectivity in the middle of a data transfer by disconnecting the Director host from the network. Execute 'nrget' command against the 'RunHist' token. Verify the partial data load was detected and removed from the database.

Results: A site verified the transfer and accuracy of the data into the relational database. The site also verified the detection of a partial data transfer and its removal from the database.

Test 7: (see Append A, para 2.3.1) Verify that the NSX sensor can send events to multiple Directors. Configure an NSX to simultaneously send alarms to multiple Directors. Once configured, generate an alarm and verify that it is received at both sites.

Results: All participating sites had their NSXs forward their event information to the central SPOCK Director at COACT satisfactorily. Event information from a specific site's NSX was monitored at central SPOCK Director at COACT in addition to its own Director.

Test 8: (see Append A, para 2.3.2) Verify that NSX systems can be configured to send alarms of different levels to a Director. Configure one NSX to send level 4 and higher alarms to a Director. Leave the default setting in place for a second NSX. Force these systems to generate alarms and verify that alarm notifications of different levels were sent to the Director.

Results: During the orchestrated tests, all NSX's reported all alarm status to the Director at COACT. Other Directors were configured to accept and report alarms at a predetermined level, varying with different sites as the scripts dictated. All worked satisfactorily.

Test 9: (see Append A, para 2.3.3) Verify Director to Director communication. Configure a Director to forward alarms from an NSX to a second Director using the 'DupDestination' entry. Force the NSX to generate alarms and verify their arrival at both Director systems.

Results: During the pre-checkout phase of the seven site connectivity, all NSX to Director, and NSX to multiple Director connections were validated. All six site Directors were dual homed to the seventh at COACT, and BCBL was also dual homed to NSA .

Test 10: (see Append A, para 3.1.1) Verify that a Director collects and displays information from multiple NSX systems in near real-time. Direct multiple attacks against several NSX systems. Verify that the attack information has been properly logged in the Director's flat files, Compare their times of the alarms, which are when the NSX detected them, with the time the alarms were entered in the flat files. This will show the propagation and processing delays for the NetRanger system at any given time.

Results: As non-intrusive attacks were launched from COACT to the other six sites, the NSX responses, and then the sites Director responses were monitored at COACT. Quantitatively, the expectations of the participants were met. When connectivity throughputs allowed, responses for router configurations were within 3 seconds, and 4 seconds for bridge configured architectures. Each site' responses seemed to vary from test to test as to their position in the COACT monitoring queue, indicating the lag varied undoubtly to the connectivity traffic flow. The claim of near-real time response was met by any reasonable definition, tending to be within a three to five second window.

Test 11: (see Append A, para 3.2.5) Verify that the Director can provide simultaneous displays. Open multiple read-only instances of the network map and set up several read-write instances of the Director. Verify that they conform to standard OpenView functionality.

Results: The COACT Director/network monitor successfully monitored all six sites NSXs and Directors during the overall test.

5.3.1 Encryption

Test 1: (see Append A, para 2.2) Verify that the NSX sensor communicates securely with the Director. *Note: No additional testing of the BorderGuard Virtual Private Network (VPN) is required. NetSentry/DPF provides the equivalent functionality of the Packet Control Facility (PCF) which was tested under the 29 March 1996 SPOCK evaluation 010504, entitled Network Security Router, Performance and Security Test.*

Results: The seven site network was brought up using the router encryption to provide encrypted sleeves between all Directors and NSX's. Encryption was verified by NSA at their site using a LAN Analyzer to verify the packets unintelligability.

Test 2: (see Append A, para 1.2) Verify the throughput of all three types of NSX systems with and without the default filters shipped with each NSX. Deploy an NSG BorderGuard 1000 and an NSX. Validate that it supports network traffic up to 512 Kbps. Deploy an NSG BorderGuard 2000™ and an NSX. Validate that it supports Ethernet (10 Mbps) as well asT1 (1.54Mbps) levels of network traffic. Deploy a Passport™ and an NSX. Validate that it supports network traffic from Fiber Distributed Data Interface (FDDI) and *T3 speeds up to 100 Mbps.*

Results: During the pretest, the BorderGuard 1000™ was based at COACT, while four sites were configured in bridge mode and two in router mode using BorderGuard 2000s™. The Center for Integrated Intelligence Systems (Space and Naval Warfare Systems Command) used the PassPort™ Security Device and a BorderGuard 1000™. The participants verified the support for the ethernet (bridge mode). Five sites connected their NSX sensors to their Directors via their local ethernets, and WAN connectivity was accomplished using T1 links. The sixth site, (BCBL) was limited by virtue of a 256 kbyte pipe, shared with all other users located at Ft. Gordon. The seventh site was connected to the other six via a 64 kbyte Integrated Services Digital Network (ISDN) link to the Internet. All worked in a satisfactory manner. The participants were *not able* to verify the FDDI and T3 claim because the required bandwidth connectivity was unavailable.

Clarification: During the Bridge mode verification tests, those sites with less than five hosts protected by the NetRanger were not able to demonstrate the ping sweep attack response. The alarm threshold was set to respond to a minimum of five ping requests to eliminate false positive indications to the Directors.

The BorderGuard 2000 routers configured as bridges in this test employ "learning bridge" algorithms which prevent the router from forwarding data across the bridge to non-existent hosts. As a result of this bridging technique, the ping requests directed towards non-existent hosts were not passed through the bridge and did not result in an alarm.

Practically speaking, a protected network will likely have the minimum (five) hosts in its configuration. The user can always lower the threshold of that particular alarm signature, bearing in mind the possibility of inadvertent false positive alarms being generated from normal maintenance actions. This is only an issue in bridge mode and only for ping sweep attacks. It does not affect the ability to detect and respond to other network attacks. It cannot be corrected by revision of the NetSentry™ or NetRanger products as it is a side effect of the layer 2 protocol.

5.4 Conclusions

In the true sense, this suite of tests proved the viability of Real-Time Network Intrusion Detection and Response for implementation today, in a warfighter networked environment. This was accomplished in a robust and effective fashion using a combination of vendor support and government operational and technical personnel. The talent and interest of no less than six military commands, two DoD Agencies, and four commercial companies participated in this event. The granularity of the WheelGroup and StorageTek Network Systems Group suite of products, (in combination with SunSparc/Ultra platforms, etc.) is further supported by laboratory tests conducted by the AIR FORCE INFORMATION WARFARE CENTER (AFIWC). For further information you may directly request their report, titled:

PRODUCT SECURITY ASSESSMENT OF THE NETRANGER INTRUSION DETECTION MANAGEMENT SYSTEM, Version 1.1, February 1997

Prepared by the Air Force Information Warfare Center/Engineering Analysis Directorate (AF/WC/EA) at Kelly Air Force Base TX.
Distribution authorized to U.S. Government Agencies only. Refer other requests for this document of AFIWC/EASM. 250 Hall BLVD., Suite 139, San Antonio. TX 78243-7063.

The rich suite of practical and needed Real-Time Network Intrusion Detection features is supported by this report.

The robustness of the architecture is evidenced by the orchestration of the fielding and connecting of this seven site inter and independent (as desired) network within a week, including pretest checkout. The few remaining configuration and connectivity problems during the actual test phase were successfully resolved using technical exchanges between the sites via public telephone, with information on the health and status provided by the architecture through the Internet.

5.4.1 FIWC SeaWitch

Though not included in the test scripts, FIWC personnel applied a penetration tool called SEAWITCH against the NetRanger system at their site.

SEAWITCH is a penetration tool developed by LCDR L. Dean Rich, USNR, designed to check an IP network or domain name for 'alive' systems to test, and then check those systems for common vulnerabilities. SEAWITCH is a tool similar to SATAN except it will actively try and access password files whenever possible. NetRanger identified many of the components of the SEAWITCH penetration attempt. SEAWITCH could not penetrate the FIWC Test system even though NetRanger did not identify a SEAWITCH-specific attack signature.

FOR OFFICIAL USE ONLY

<u>SECURITY PROOF OF CONCEPT KEYSTONE</u>

NETRANGER REAL-TIME NETWORK INTRUSION DETECTION
Performance and Security Test
Appendix A
Scripts Used During Testing

Prepared for:
Maryland Procurement Office
9800 Savage Road
Fort George G. Meade, Md. 20755

Contract:
MDA904-96-C-0215

By
COACT, Inc.
9140 Guilford Road, Suite L
Columbia, Maryland 21046

Document No. 010511

30 April 1997

Not releasable to the Defense Technical Information Center Per DoD Instruction 3200.12 Distribution limited to U. S. Government Agencies Only. This document contains NSA information (Jan.1992). Request for this document must be referred to the Director NSA.

FOR OFFICIAL USE ONLY

Appendix A

Introduction

This Appendix contains information related to the SPOCK (Security Proof of Concept Keystone) report on the WheelGroup NetRanger Real-Time Network Intrusion Detection Product.

Content. The following materials are presented in this Appendix:

1) Event Log,

2) Scripts used during testing,

3) Event-to-IP Address Matrix for NetRanger Evaluation,

4) NetRanger Demonstration Group Legal Release Request.

Event Log

The Event Log contains a record of the test scenarios that were performed, the participants in each test event, the claims tested, and the keywords for the test procedures. The Event Log is presented in table format. The rows and columns in this log are defined as follows:

1) The left-hand column contains the claims and the paragraph references to the test scripts.

2) The top row contains the keywords for the test procedures.

3) The bottom row contains identifiers for the participants in each test event.

Master event numbers are also presented in bold at the bottom of the Event Log; these numbers were used to check off each test event as it was verified. Each of the participants documented the results of the tests as they were performed. See Appendix B, Participant Verification and Comments, for the actual comments from the participants.

Scripts used during testing

All of the test scripts that are referred to in the 'Event Log' are also included in this Appendix. Each test scenario outlines the goal of the test, a manual reference, and the procedures to be followed during the test.

Event-to-IP Address Matrix for NetRanger Evaluation

The Event-to-IP Address Matrix for NetRanger Evaluation document is a record that was used to verify each IP destination prior to executing a test script. This evaluation significantly minimized the risk of misreading non-intrusive network attacks.

NetRanger Demonstration Group Legal Release Request

The NetRanger Demonstration Group Legal Release Request was distributed to all of the participants in the SPOCK testing. Full approval was received by each organization participating in these activities. The legal release request is included in this appendix to demonstrate that this process was performed.

	Deploy as both Bridge and Router	Net Configuration will Demonstrate	Alarms will Reflect Participant's Requirement	Net Configuration will Demonstrate	Net Configuration will Demonstrate	SPOCK Director will Demonstrate	Run Level 5 Attack	Reconfigure and run same attack, showing different Alarm Level	Verify NSX Configuration set for Level 1 ERRORS and COMMANDS	Intentionally Misconfigure and Show Error Logged	Confirm Specific Action is Allowed	Remove Previous Action from Director's Authorization and Show Failure to Execute Specific Action	Disrupt Connection Between Director and NSX, Send Attack to NSX	Reconnect Director and View Queued Alarm Messages
Claims proven by Static Network Configuration														
Deployed as Bridge or Router (1.1)	XXX													
Can Be Broadcast to Multiple Locations (2.3.1)		XXX												
Alarms Notification is Highly Configurable (2.3.2)			XXX											
Information can be Propagated Between Directors (2.3.3)				XXX										
Real-Time Collection and Display from Multiple NSXs (3.1.1)					XXX									
Provides Centralized Support (SPOCK Monitor) (3.2.5)						XXX								
Phase I														
Configurable Event Severity Levels (1.4)							XXX	XXX						
System Auditing (1.6)									XXX	XXX				
Command Authorization (1.5)											XXX	XXX		
Connection Oriented Communication (2.1.1)													XXX	XXX
Fault-Tolerant Communication (2.1.2)														
Remote NSX Management (3.1.2)														
Allows Operator Response (3.1.3)														
Define Display Thresholds (3.2.2)														
Minimize Screen Clutter (3.2.3)														
Hierarchical Propagation of Alarms (3.2.4)														
Notify Off-Duty Personnel of Events (3.3)														
Phase II														
Context-Based Signatures (1.3.1.1)														
Content-Based Signatures (1.3.1.2)														
Composite Signatures (1.3.1.3)														
Automatic Shunning (1.3.2)														
Logging (1.3.3)														
Automatic Log File Management (1.7)														
Provides Description of Security Events (3.2.1)														
Stages Data to Relational Database (3.4)														
Key: 1 = AFWIC, 2=LIWA, 3=NSA, 4=BCBL, 5=FIWC, 6=SPAWAR, 7=SPOCK (COACT)	Proven by Static Network Configuration	Proven by Static Network Configuration	Proven by Static Network Configuration	Proven by Static Network Configuration	Proven by Static Network Configuration	Proven by Static Network Configuration	2,5,7	2,5,7	3,6,7	3,6,7	2,3,7	2,3,7	1,5,7	1,5,7
Event Number	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*

Event Number	Key: 1 = AFWIC, 2=LIWA, 3=NSA, 4=BCBL, 5=FIWC, 6=SPAWAR, 7=SPOCK (COACT)	Stages Data to Relational Database (3.4)	Provides Description of Security Events (3.2.1)	Automatic Log File Management (1.7)	Logging (1.3.3)	Automatic Shunning (1.3.2)	Composite Signatures (1.3.1.3)	Content-Based Signatures (1.3.1.2)	Context-Based Signatures (1.3.1.1)	Phase II		Notify Off-Duty Personnel of Events (3.3)	Hierarchical Propagation of Alarms (3.2.4)	Minimize Screen Clutter (3.2.3)	Define Display Thresholds (3.2.2)	Allows Operator Response (3.1.3)	Remote NSX Management (3.1.2)	Fault-Tolerant Communication (2.1.2)	Connection Oriented Communication (2.1.1)	Command Authorization (1.6)	System Auditing (1.5)	Configurable Event Severity Levels (1.4)	Phase I		Provides Centralized Support (SPOCK Monitor) (3.2.5)	Real-Time Collection and Display from Multiple NSXs (3.1.1)	Information can be Propagated Between Directors (2.3.5)	Alarms Notification is Highly Configurable (2.3.3)	Can Be Broadcast to Multiple Locations (2.2.1)	Deployed as Bridge or Router (1.1)	Claims proven by Static Network Configuration	
15	3,7																	XXX														Verify Primary Comms Path, and Disconnect
16	3,7																	XXX														Verify Secondary Comms Path
17	3,5,7																XXX															Perform *getbulk* Command to Multiple NSXs from Single Director
18	3,5,7																XXX															Alter Configuration of Multiple NSXs from Single Director
19	1,2,3,4,5,6,7															XXX																Operator Applies *SHUN* to ongoing attack
20	3,4,5,7														XXX																	Attack 3 NSXs to produce Red, Yellow and Green Icons
21	3,4,6,7														XXX																	Reset Thresholds to Produce all Yellow Icons When Same Attack Set Run Second Time
22	1,2,3,4,5,6,7													XXX																		Run attack signature 4 or 5 times to all sensors
23	1,2,3,4,5,6,7													XXX																		Operators see that multiple attacks have been automatically consolidated into a single icon
24	1,2,3,4,5,6,7												XXX																			Run level two attack signature
25	1,2,3,4,5,6,7												XXX																			Run level 5 attack signature and view propagation of Director display hierarchy
26	3,5,7											XXX																				Configure Director to generate automatic email and/or pages
27	3,5,7											XXX																				Run attack and observe email and/or pager activation
28	1,2,3,4,5,6,7								XXX																							Attempt *rlogin* connet to a protected host from a machine authorized access to the network and observe alarm generated

	Execute *sendmail* attack and observe alarm generation	Attempt to send email with content identified as a policy violation and observe alarm generation	Telnet into target host and generate *atomic* alarm	Run port sweep against host and observe *composite* alarm	Run SATAN in heavy mode and observe corresponding *composite* alarm	Run SATAN in light mode and observe corresponding *composite* alarm	Set NSX to autoshun telnet attempt. Attempt telnet and observe autoshun occur	Attempt telnet and operator manually respond with shun	Attempt telnet through sniffer and observe autoshun occurs within 3 seconds	Configure NSX to log keystrokes from specified attack type	Run attack and verify that keystroke capture occurs	Configure NSX to dump Event and IP Session logs to an archive device	Verify that data has transferred to off-line device	Run Attack to generate alarm
Claims proven by Static Network Configuration														
Deployed as Bridge or Router (1.1)														
Can Be Broadcast to Multiple Locations (2.3.1)														
Alarms Notification is Highly Configurable (2.3.2)														
Information can be Propagated Between Directors (2.3.3)														
Real-Time Collection and Display from Multiple NSXs (3.1.1)														
Provides Centralized Support (SPOCK Monitor) (3.2.6)														
Phase I														
Configurable Event Severity Levels (1.4)														
System Auditing (1.5)														
Command Authorization (1.6)														
Connection Oriented Communication (2.1.1)														
Fault-Tolerant Communication (2.1.2)														
Remote NSX Management (3.1.2)														
Allows Operator Response (3.1.3)														
Define Display Thresholds (3.2.2)														
Minimize Screen Clutter (3.2.3)														
Hierarchical Propagation of Alarms (3.2.4)														
Notify Off-Duty Personnel of Events (3.3)														
Phase II														
Context-Based Signatures (1.3.1.1)														
Content-Based Signatures (1.3.1.2)	XXX	XXX												
Composite Signatures (1.3.1.3)			XXX	XXX	XXX	XXX								
Automatic Shunning (1.3.2)							XXX	XXX	XXX					
Logging (1.3.3)										XXX	XXX			
Automatic Log File Management (1.7)												XXX	XXX	
Provides Description of Security Events (3.2.1)														XXX
Stages Data to Relational Database (3.4)														
Key: 1 = AFWIC, 2=LIWA, 3=NSA, 4=BCBL, 5=FIWC, 6=SPAWAR, 7=SPOCK (COACT)	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	1,2,3,4,5,6,7	3,5,7	3,5,7	1,2,3,4,5,6,7
Event Number	*29*	*30*	*31*	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*	*40*	*41*	*42*

	Highlight alarm and note information displayed	Verify that data has been staged to a relational database IAW procedure 1 of claim 3.4	Verify that a Director can detect staging errors and rolls back an incomplete transfer IAW procedure 2 of claim 3.4
Claims proven by Static Network Configuration			
Deployed as Bridge or Router (1.1)			
Can Be Broadcast to Multiple Locations (2.3.1)			
Alarms Notification is Highly Configurable (2.3.2)			
Information can be Propagated Between Directors (2.3.3)			
Real-Time Collection and Display from Multiple NSXs (3.1.1)			
Provides Centralized Support (SPOCK Monitor) (3.2.5)			
Phase I			
Configurable Event Severity Levels (1.4)			
System Auditing (1.5)			
Command Authorization (1.6)			
Connection Oriented Communication (2.1.1)			
Fault-Tolerant Communication (2.1.2)			
Remote NSX Management (3.1.2)			
Allows Operator Response (3.1.3)			
Define Display Thresholds (3.2.2)			
Minimize Screen Clutter (3.2.3)			
Hierarchical Propagation of Alarms (3.2.4)			
Notify Off-Duty Personnel of Events (3.3)			
Phase II			
Context-Based Signatures (1.3.1.1)			
Content-Based Signatures (1.3.1.2)			
Composite Signatures (1.3.1.3)			
Automatic Shunning (1.3.2)			
Logging (1.3.3)			
Automatic Log File Management (1.7)			
Provides Description of Security Events (3.2.1)	XXX		
Stages Data to Relational Database (3.4)		XXX	XXX
Key: 1 = AFWIC, 2=LIWA, 3=NSA, 4=BCBL, 5=FIWC, 6=SPAWAR, 7=SPOCK (COACT)	1,2,3,4,5,6,7	1	1
Event Number	***43***	***44***	***45***

NetRanger™ Capabilities and Test Scenarios
WheelGroup Corporation

Version 1.2

24 February 1997

Table of Contents

This document presents *Test Scenarios* for evaluating WheelGroup's NetRanger Network Secur-ity Management System, and is based upon capabilities described in Chapter I of the *NetRanger User's Guide.* These scenarios are presented within the context of NetRanger's three major systems: **NSX, Communications,** and **Director.** Please refer to the *NetRanger User's Guide* for a more complete overview of these capabilities. We also recommend that you read through this entire document before proceeding with any tests, because many of the tests make forward references to features described in later sections.

1. NSX Capabilities

1.1 Can Be Deployed as a Bridge or a Router

NetRanger's remote NSX systems consist of one or more packet filter devices that copy data to an NSX sensor. Packet filter devices currently supported by NetRanger include the BorderGuard and Passport systems from StorageTek, Network Systems Group (NSG). While architecturally different, these devices use StorageTek's **NetSentry** filtering language which provides a standard data format for NetRanger. In addition to being able to copy information to an NSX sensor, these packet filters can be deployed as bridges or routers.

Bridge mode operation provides two basic benefits over routing:

Operational
An NSX system can be deployed on a network without disrupting existing router and switch configurations. This greatly simplifies deployment of NetRanger in large established enterprises.

Security
The system is invisible on the network because it is a "bump on the wire" without the need for subnetting. Only one network address is necessary. This low visibility adds to the overall security of the system.

Test Scenario

Goal:

Verify that the NSX can operate as a bridge or a router.

Reference:

Chapter II in the *NetRanger User's Guide.*
BorderGuard User's Guide

Procedure:

1) Deploy an NSG BorderGuard as a bridge device. Note that the BorderGuard's filters can be configured to "hide" the NSX on the network.
2) Deploy an NSG BorderGuard as a router.

1.2 Handles Network Throughputs from 56 Kbps up to 100 Mbps

No other network Intrusion Detection System (IDS) covers as broad a range of network bandwidths as NetRanger. The **NSX 1000** handles network throughput up to 512 Kbps and is targeted at customers with ISDN Internet connections. The **NSX 2000** is designed to scale up to Ethernet speeds (10 Mbps) and is intended for T1 network connections. The **NSX 5000** is designed to scale up to 100 Mbps and it is intended for T3/DS3 and FDDI connections. All three NSX systems contain two basic components: an NSG *packet filter device* and an NSX *sensor.* All communication between these two components is managed via NSG's **NetSentry** operating system.

Test Scenario

Goal:

Validate the throughput of all three types of NSX systems with and without the default filters shipped with each NSX.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure:

1) Deploy an NSG BorderGuard 1000 and an NSX. Validate that it supports network traffic up to 512 Kbps.
2) Deploy an NSG BorderGuard 2000 and an NSX. Validate that it supports Ethernet (10 Mbps) as well as T1 (1.54 Mbps) levels of network traffic.
3) Deploy an NSG Passport and an NSX. Validate that it supports network traffic from FDDI and T3 speeds up to 100 Mbps.

Initial tests should be performed without any filters applied. Filters can then be applied to an NSX system's BorderGuard or Passport device by connecting to the NSX via *telnet* and running */usr/nr/bin/nrconfig*.

NOTE: This type of test is usually based upon generation of artificial network traffic. Test results will be misleading unless this test traffic is based on a mixture of "typical" events, such as *telnet*, *ftp*, e-mail, and WWW activity. Flooding the network with large volumes of ping or TCP connect request packets will generate spurious results. Network traffic should also be based on a normal sized packets, such as 64 or 1514 bytes. Refer to the *NetRanger User's Guide* for complete configuration information.

1.3 *Performs Real-Time Intrusion Detection, Notification, and Response*

NetRanger is built around the concept of Permissive Networking™, which allows authorized network traffic to flow freely past an NSX system until a *pattern of misuse* is detected, at which time three things can occur:

- **the user is notified of the event** on one or more Director systems.
- **the attack is shunned.**
- **the event is logged.**

Misuse is defined as suspicious as well as unauthorized activity. While many network security solutions, such as firewalls, protect against known vulnerabilities by erecting static barriers, these types of security devices typically cannot detect misuse without disrupting other network activity.

NetRanger uses expert system rules to ***identify*** and ***respond*** to patterns of misuse in ***real-time***. These rules are based on known security policy violations, external access techniques, context-based signatures, and content-based signatures. The security policy enforced by an NSX system depends on how the rules for each of these capabilities are configured. The default configurations for an NSX can be found in Chapter III of the *NetRanger User's Guide.*

Please note that:

- Although an NSX responds to a pattern of misuse in real-time, *the point at which that pattern is detected depends on the attack signature.* Attacks based upon single actions, such as an *address ping*, elicit a response immediately. Others, such as a SATAN attack, are not acknowledged until the NSX detects that a certain number of ports have been accessed on a specific host. This has a direct impact on how quickly alarms are generated and appear at a Director. Before this type of testing begins, technicians should have a basic understanding of all of NetRanger's signature types, which are discussed in Chapter III of the *NetRanger User's Guide*.
- Notification of an attack is generated in less than a second, but *it may take up to three seconds for the NSX to shun that attack.* The time it takes to apply a shun filter

depends on the size of existing filters and the volume of traffic passing through the packet filter. In most cases an attack is shunned in a fraction of a second.

- The time required for *remote notification* and logging on a Director system *is largely a function of network loads.*
- An NSX inspects *outgoing* as well as *incoming* network traffic. In addition to facilitating more complete reconstruction of events, this ***bi-directional*** capability enables the NSX to detect *internal* as well as *external* threats to an enterprise.

1.3.1 Alarm Signatures

NetRanger categorizes patterns of misuse into two basic types of network signatures: ***context*** and ***content***. Context-based signatures deal with *transmission state* (defined by the structure of packet headers), while content-based signatures focus on *what* is being transported (-- the binary/ASCII data). Many signatures also tend to be based on a ***composite*** sequence of events. Test scenarios are provided for each type of alarm signature.

1.3.1.1 Context-Based Signatures

An NSX detects patterns of misuse and generates corresponding alarms in *real-time*. This section describes how to establish a security policy and generate alarms for context-based attacks.

Test Scenario

Goal:

Verify that an NSX sensor can detect a context-based attack and generate an alarm either when:

1. someone tries to access a ***unauthorized service*** on a host protected by an NSX system,

or

2. an ***unauthorized external system*** tries to access the protected network.

Before you begin you must verify that the Director and NSX systems are talking to each other properly. As the *NetRanger User's Guide* explains, communication between NSX and Director systems is based upon a proprietary, fault tolerant, point-to-point network protocol. The easiest way to verify a connection is to run the command */usr/nr/bin/nrconns* on both systems.[1] Each system should return an entry for the other system with a status of *[Established]*. An NSX configured with an alternate as well as a primary Director connection would generate output similar to the following:

```
Director.TestSys   Connection 1:10.1.64.2         45000   1   [Established]
         sto:0009 with Version 1
Director.TestSys   Connection 2: 209.198.133.5  45000   1   [Listen]
         sto:0009
```

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure 1:.

[1] Refer to the section 3.1.2 *(Provides Remote NSX Management)* for instructions on how to retrieve this information remotely via the **nrConfigure** interface.

You can commence with the alarm tests once the Director-NSX connection has been verified. For example, *rlogin* is disabled in the default configuration shipped with an NSX system. An alarm should be displayed on the Director security map when you try to *rlogin* onto a host protected by the NSX from a host that resides outside the protected network.

Procedure 2:

While an NSX system can be configured to automatically shun an attacking host (as described in section 1.3.2) an NSX system can also be configured to implement a site's security policy by **permanently** blocking specific hosts or networks.

The following test procedure shows how to permanently block an external host on a Borderguard and to configure an NSX system to generate an alarm when an unauthorized external host tries to access a protected host.

Once an unauthorized external host has been identified, *telnet* onto the NSX and insert the host's IP or network address into the filter **LAST_BLOCK_FAIL** that resides in the file **last.fil** in */usr/nr/etc/nsc/templates.* The filter should then look something like the following:

```
filter LAST_BLOCK_FAIL
        ip_sa in (ATTACK_IP_ADDRESS)
                copy_to NSX_ADDRESS 35399
                icmp_unreachable fail;
end
```

Transfer this filter onto the BorderGuard and compile it. Then apply it on the last filter point using the following command:

```
ip apply LAST_FILTER on last
```

Now try to *telnet* from the unauthorized host onto a protected host. An **ICMP Unreachable** alarm (Sigid 2001) should then appear on the Director security map.

1.3.1.2 Content-Based Signatures

In addition to detecting misuse based on packet header information, NetRanger detects misuse based on the *content* of network packets. These patterns are based upon *user-defined* keywords and text strings. These keywords and phrases can also be linked to specific host *services*. For example, detection of the phrase "Company Confidential" can be limited to outgoing e-mail (port 25). As the *NetRanger User's Guide* explains, this feature has been implemented in ways that minimize *false positives*: regular expression syntax, detection of duplicate events, etc. Please note that content-based signatures generate the same response capabilities as context-based signatures.

Test Scenario

Goal:

Verify that the NSX sensor can:

1) by default detect and produce alarms for many standard user attacks.
2) detect and produce alarms for user-defined content-based signatures.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure 1:

Review the default signatures and alarm levels documented in the *NetRanger User's Guide*. Choose several for testing. For example, execute part of a common sendmail type attack. *Telnet* to port 25 (the mail port) of the authorized mail server within the protected network. Type "VRFY" or "++" and verify that an alarm is generated. Repeat these attacks a second time, but add spurious information, such as upper and lower case characters, intervening spaces, back-spaces, and so on (e.g., "Vr ^Hfy"). Verify that alarms are still generated for these attacks.

Procedure 2:

Create a *sensord* **RecordOfStringName** entry for a specific TCP service and then define a matching **SigOfStringMatch** alarm entry. Test the system by using that keyword against that TCP service and verify that it was detected. Depending upon your configuration, this event can be confirmed via inspection of a log file as well an alarm notification on a Director console.

1.3.1.3 Composite Signatures

In order to detect a number of different context-based attack signatures, an NSX must interpret *sequences of events*. A SATAN attack (signature 6001) is one of the better known examples of a *composite* signature, while an ICMP Echo Request (signature 2004) is an example of an *atomic* signature.

Test Scenario

Goal:

Verify that the NSX sensor can detect and produce an alarm for composite signatures.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure:

Telnet into a target host and type "IFS=/" followed by the Enter/Return key[2]. An *atomic* alarm should be generated. Initiate a port sweep against the same host and a *composite* alarm should be generated. Run SATAN against the network and several *composite* alarms should be generated, including one that indicates whether SATAN was run in *heavy* or *normal* mode.

1.3.2 Shunning Network Connections

An NSX can be configured to automatically shun a network connection when it detects that a host on its protected network is the source of an attack. This is accomplished by dynamically changing the **NetSentry** filter policy on the BorderGuard or Passport security device. As noted earlier, while shunning a connection usually occurs within a fraction of a second from the time an attack is detected, it can take up to 3 seconds on a system that

[2] IFS=/ is a standard component of several different types of attacks used to gain "root" privileges on target hosts.

contains complex filters and is monitoring large volumes of traffic.

Other noteworthy features of NetRanger's shunning include:

- An attacking host can be shunned without disrupting network traffic between other hosts.
- The duration of a shun is configurable, thereby allowing the communication to be reestablished after a set time period. The barrier is not reestablished unless the threat reappears.

Dynamic shunning is a core feature of NetRanger's Permissive Networking™ technology and represents a dramatic advance over more static network security solutions, such as proxy-based firewalls, which erect barriers to protect systems from attack.

Test Scenario

Goal:

1. Verify that an NSX sensor can automatically shun a connection.
2. Verify that an NSX sensor can be manually configured to shun an attack.
3. Verify that the source of an attack can be shunned within three seconds.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure 1:

Identify a simple attack against a specific service, such as the appearance of the string "IFS=/" in a *telnet* session. This content-based signature has a *sensord* **RecordOfStringName** identifier of 301. Configure the NSX to shun this type of attack by setting this signature's **SigOfStringMatch** *action* field to "1". Note that *shun duration* is specified by the *sensord* **MinutesOfAutoShun** token. The default timeout is 15 minutes.[3]

This type of attack is relatively easy to run. *telnet* onto a *target* host protected by an NSX from an *attacking* host that sits *outside* the target network. Then enter the string "IFS=/" (followed by Enter/Return) from the *telnet* session. This event should register almost immediately on your Director system. Notification can also be confirmed by searching for a *signature_id* of 8000 with a *subsigna-ture_id* of 301 in the current log file in the Director's current */usr/nr/var/log.<date>* file.

The simplest way to verify that the shun has been applied is to do the following:

1) Before launching the *telnet* attack, open a separate terminal session on the attacking host and initiate a **ping** command against the target host. This will cause the ping command to generate a ping every second against the target host.
2) Note the status of the ping command when you press the Enter/Return key for the IFS=/ *telnet* attack.
3) The ping requests should fail at about the same time as the notification is displayed the Director.

Procedure 2:

Shunning can also be manually initiated by hi-lighting the alarm icon and then selecting the **Security->Shun** menu option from the Director. Verify that the attacking system can no longer access any of the hosts on the protected network by repeating the attack. Please note that the shun can be manually disabled at any time by hi-lighting the alarm icon and then selecting **Security->Unshun**.

[3] Note that an NSX cannot shun a host address if it is listed as a *sensord* **RecordofExcluded-Address.**

Procedure 3:

In order to confirm the amount of time it takes to shun an attacking host do the following:

1) Install a sniffer between the target and attack hosts. Ideally the sniffer should be as close to the target host as possible.
2) Initiate a **ping -s** against the target host from the attack host.
3) Configure the sniffer to monitor ICMP packets from the target host.
4) Run the IFS=/ attack previously described.
5) Compare the time stamps of the last successful ping against the first ACK returned by the target host. This should provide you with a rough estimate of how long it took the NSX system to shun the attacking host.

1.3.3 Logging

An NSX can log *session activity* as well as the events that make up a pattern of misuse. Session logging is often described in the security community as "keystroke capture." This is accomplished, once the pattern of misuse is detected, by dynamically opening a log file on the NSX's hard drive and copying the IP session information.

Test Scenario

Goal:

Verify that the NSX sensor initiates a session log based on detection of misuse

Reference:

Chapter III in the *NetRanger User's Guide.*

Procedure:

Identify a simple attack against a specific service, such as a Smail attack (signature id of 3100). Configure the NSX to log keystrokes for that signature by resetting its **SigOfGeneral** *action* field to "2". Any time that type of attack is detected all incoming and outgoing keystrokes will be logged to a file named *iplog.<src IP address>* in */usr/nr/var/iplog.* Verify the keystroke content of the log file by running it through the command */usr/nr/bin/transcript.*

1.4 Event Severity Levels are Configurable

The NetRanger system is shipped with default alarm "level" settings for each alarm type. As explained in the *NetRanger User's Guide,* these alarm levels can be reassigned by the user on an NSX-by-NSX basis. For example, the default alarm level generated for a failed *rlogin* is '3'. While this may be appropriate for Site A, an administrator may want to raise the level to '4' for this signature at Site B. Based on these configurations, a Director would receive *rlogin* alarms of level '3' from Site A and of level '4' from Site B, thereby allowing a technician to assess their relative significance on a site-by-site basis.

Test Scenario

Goal:

Verify that the NSX sensor can be configured to send user-defined alarm levels

Reference:

Chapter III in the *NetRanger User's Guide.*

Procedure:

Review NetRanger's default alarm level configurations and choose several to test. For example a SATAN attack generates a default level 5 alarm. Run a SATAN attack and verify a level 5 alarm is generated. Then change the default to level 4 and repeat the process. This can be tested across multiple NSX systems. Note that alarm levels set **below** the Director's OpenView alarm threshold will not be displayed. They will, however, be recorded in the log files of the NSX which is generating those alarms. Also note that if they are *below* the Director's Open-View alarm threshold, but *above* the minimum alarm level to send to the Director, the Director will continue to send these alarms to any duplicate destinations specified in its */usr/nr/etc/smid.conf* file.

1.5 *System Activities are Audited*

In addition to logging *network events* pertaining to patterns of misuse, NetRanger also logs *commands* and *errors* generated by the NSX itself or by users. These log entries provide important information ***when trying to reconstruct operational events.***

Test Scenario

Goal:

Verify that the NSX sensor maintains a log of all errors and commands

Reference:

Chapter II in the *NetRanger User's Guide*.

Procedure:

Verify that at least one destination in an NSX's */usr/nr/etc/destinations* file is configured for level 1 ERRORS and COMMANDS. Issue a series of commands from the Director and verify that they are logged into the appropriate *destination's* log file (typically on the same Director). Then misconfigure one of the applications on the NSX to generate an error and verify it is logged. For example, change *managed's* ROUTER_PASSWORD. When *managed* is unable to log into the router, it will generate an error message.

1.6 *Command Authorization is Configurable*

Secure communication between NSX and Director systems is controlled by two mechanisms: encrypted sleeves and access control lists. Encrypted sleeves are provided by StorageTek's Virtual Private Network (VPN) facility, which is presented in section 2.2. *Access* is tightly controlled via NetRanger's proprietary access control list facility. These lists identify the authorized hosts and corresponding services they can perform: set, unset, get, getbulk, and exec. These services are equivalent to standard Unix read, write, and execute commands. Please note that while an NSX can be deployed without encrypted sleeves, access control lists are required.

Test Scenario

Goal:

Verify that access to an NSX sensor is dictated by its access control list.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure:

Review an NSX's default authorizations (*/usr/nr/etc/auths*). This file specifies the actions each Director is allowed to submit to the NSX. Confirm that a given action (such as *getbulk*) is allowed by bringing up **nrConfigure** and executing that command against a service token (such as *postofficed's* **DestinationConnectionStatus**). Remove *getbulk* from the Director's authorization on the NSX and repeat the prior steps. The command should fail the second time. Reinstate the Director's *getbulk* authorization and resume testing.

1.7 *Automatic Log File Management Simplifies System Maintenance*

The logging of network events represents a core NetRanger capability. If the log files generated by an NSX sensor are not managed properly, however, the operational integrity of an NSX sensor could be compromised. Every NSX sensor is shipped with the SAP subsystem, which can be configured to automatically compress files and dump them to an secondary data store, such as a tape device. While the steps required to *verify* this capability are somewhat complex, the NSX SAP package is configured by default to automatically compress log files. These services are also designed to run at a lower priority to minimize their impact on the sensor's intrusion detection processes.

Test Scenario

Goal:

Verify that an NSX sensor automatically dumps Event and IP Session logs to an archive device.

Reference:

Chapter V in the *NetRanger User's Guide*.

Procedure:

1) Establish proper file management thresholds for the NSX's *sapd.*

Via **nrConfigure** execute ***nrget*** against the ***sapd*** token ***FileMgmt.*** At a minimum, the information returned to you should include the following two entries:

```
FileMgmt  VARNEW_DUMP DIRSIZE 50000  /usr/nr/var/new /usr/nr/bin/sap/ctl_dump.sh
FileMgmt  IPLOG_DUMP DIRSIZE  50000  /usr/nr/var/iplog  /usr/nr/bin/sap/ctl_ipdump.sh
```

These two FileMgmt entries define directory size thresholds for dumping Event and IP Session log files. In the preceding example the size threshold for both of the FileMgmt entries is set at 50 MB. Event logs residing in */usr/nr/var/new* are processed by the utility *ctl_dump.sh.* IP Session logs residing in */usr/nr/var/iplog* are processed by *ctl_ipdump.sh.* Please note that during testing the DIRSIZE threshold for these FileMgmt entries should be set to a lower value, such as 2 MB (2000). Use **nrConfigure** to reset this threshold by executing ***nrset*** against this token.

2) Configure *sapd's* DUMP_OFFLINE feature.

If one does not already exist, use **nrConfigure** to create a *sapd* DUMP_OFFLINE FileMgmt entry for the NSX system. The ***Optional Parameters*** should look as follows:

```
DUMP_OFFLINE  5000  /usr/nr/var/dump /usr/nr/bin/sap/ctl_offline.sh
```

This instructs *sapd* to execute the script ctl_offline.sh whenever the amount of disk space consumed by files in the directory */usr/nr/var/dump* exceeds 5 MB.

3) Define the actual command(s) to be executed by the DUMP_OFFLINE script *ctl_offline.sh*

This script contains the or commands required to move the files in */usr/nr/var-/dump* to the appropriate archive device. This script is shipped with example code for either writing to an NFS mounted directory or a tape device. Please note that you will need to *telnet* onto the NSX sensor in order to modify *ctl_offline.sh.*

1) Reduce *loggerd's* serialization thresholds to levels that facilitate testing.

Use nrConfigure to set the NSX's *loggerd* **NumberOfSwitchMinutes** to **5 minutes** and **NumberOfSwitchBytes** to **5 Kbytes** (5000). The NSX will then serialize log data as soon as either of these thresholds is exceeded.

1) Commit new configuration to disk

From **nrConfigure,** perform an ***nrexec*** against the ***WriteFileConfig*** token for both ***loggerd*** and ***sapd.***

2) Verify that events are being logged.

Verify that events are being properly logged by checking *sapd* **RunHist** and **VarSize** on the NSX sensor. For example, a **nrget of RunHist** for **sapd** might return the following:

```
3 : VARNEW_DUMP : (/usr/nr/bin/sap/ctl_dump.sh, /usr/nr/var/new)
    (ok) (02/07 10:15 , 02/07 10:45 )
```

which indicates that the *nr/var/new* directory has been archived 3 times – the first archive event occurred at 10:15 on 2/7, the last (3rd) event occurred at 10:45.

VarSize shows you the current status of all of the */usr/nr/var* directories that are used by *sapd* to manage NSX log files. A relatively simple way to monitor log file growth and archiving is to repeatedly query against these tokens via **nrConfi-gure**. This will show you how they change over time and archive processes.

1) Verify data has been successfully transmitted.

Offline archival of log data has occurred when **RunHist** returns an entry for DUMP_OFFLINE. You can then use the appropriate tools to verify that the data was transferred to the designated offline device (e.g., tar -tvf /dev/rmt/0).

2. Communications Architecture

2.1 All Communication is "Guaranteed"

2.1.1 Connection-Oriented Communication

WheelGroup has implemented a connection-oriented, point-to-point communication protocol on top of UDP/IP to facilitate large-scale deployment of NetRanger across multiple network protocols. This also allows NetRanger to span address translation gateways. Connectivity is maintained via a system heartbeat and all messages are re-sent when a destination is unable to reassemble all of a message's UDP packets.

Test Scenario

Goal:

Verify that the NSX sensor queues information when connectivity to the Director is lost and when the link is re-established that no information is lost.

Reference:

Chapter I in the *NetRanger User's Guide*.

Procedure:

Disrupt the network connectivity between an operating NSX and Director. Cause alarms to be generated that would normally be sent to the unreachable Director. Reestablish connectivity to the Director and verify that queued messages are resent.

2.1.2 Fault-Tolerant Communication

An NSX or Director can be configured with up to 32 alternate routes. These routes can be network based or on-demand dial up via a PPP link. System housekeeping services guarantee that the preferred route is used whenever possible. This feature minimizes the possibility of NetRanger communication failure due to network failures, power outages, or other disruptions.

Test Scenario

Goal:

Verify that the NSX sensor will send information to the Director via alternate routes as links are broken and re-established.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure:

Identify a primary route to one Director and a secondary route to the same Director within the test network. If necessary, a secondary route can be established to a dual-homed machine that has an alternate route to the Director. Bring the NSX on line to verify that alarms are being received at the Director. Then execute the command */usr/nr/bin-nrconns* to check the primary route of the data. The primary route to the Director will display the connection as "*[Established]*" and the secondary route will be displayed as "*[Listening]*".[4]

Example 1:

```
Director.TestSys    Connection 1: 10.1.64.2        45000   1   [Established]
```

[4] In order to run *nrconns* the */usr/nr/bin/nr.configd* daemon must be running and there must be a entry for the local NSX in the */usr/nr/etc/auths* file.

```
          sto:0009 with Version 1
Director.TestSys    Connection 2: 209.198.133.5  45000   1   [Listen]
          sto:0009
```

Break the network connection for the primary route and send several alarms. Verify that the alarms are still being received at the Director and that no alarms have been lost in the changeover. Verify the connections using the same */usr/nr/bin-/nrconns* command. You should now see the primary route displayed as "*[SynSent]*" indicating that it is trying to establish this connection, and the secondary route should be displayed as "*[Established]*".

Example 2:

```
Director.TestSys    Connection 1: 10.1.64.2        45000   1   [SynSent ]
          sto:0009 with Version 1
Director.TestSys    Connection 2: 209.198.133.5  45000   1                    [Established]
          sto:0009
```

2.2 *All Communication Is "Secure"*

Secure communication between NetRanger Director and NSX systems is accomplished via Virtual Private Networks (VPNs) that NSG is able to establish between one or more BorderGuard devices. NSG uses standard RSA public key algorithms to initialize their encrypted sleeves. The sleeve itself can be configured to work with DES, Triple DES, IDEA, or NSC1 private key encryption. NSX and Director access control is implemented in a transparent manner on top of these sleeves.

Test Scenario

Goal:

Verify that the NSX sensor communicates securely with the Director

Reference:

Chapter I in the *NetRanger User's Guide*.

Procedure:

No additional testing of the BorderGuard VPN is required. NetSentry provides the equivalent functionality of the Packet Control Facility (PCF) which was tested under the *29 March 1996 SPOCK evaluation 010504*, entitled Network Security Router, Performance and Security Test.

2.3 *NSX/Director Communications Are Configurable and Flexible*

2.3.1 Alarms Can Be Broadcast to Multiple Locations

NSX systems can be configured to send alarms to multiple Directors. This allows multiple operations centers to monitor activity across an entire security perimeter. Command and control is then dictated by an enterprise's operational procedures and protocols.

Test Scenario

Goal:

Verify that the NSX sensor can send events to multiple Directors

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure:

Configure an NSX to simultaneously send alarms to multiple Directors. If you don't have two Directors available you can send alarms to the *loggerd* daemon on another NSX machine. Once configured, generate an alarm and verify that it is received at both sites.

2.3.2 Alarm Notification Is Highly Configurable

The NSX is configured to send alarms to a Director *based on alarm level.* By default, alarms of level 3 and higher are automatically sent out to a Director. Lower alarm levels (1-2) can be requested on demand, and in some instances it may be advantageous to have one or more high-profile NSX systems configured to transmit lower level alarms. WheelGroup recommends against having all of the NSX systems on a security network continually transmit alarms below level 2, however. The reasons for this are as follows:

1. It can have an adverse impact on network throughput during peak loads.
2. It can lead to excessive alarming on a Director platform. This frequently makes it more difficult to assess what is actually going on and can lead to the "cry wolf" syndrome.

Test Scenario

Goal:

Verify that NSX systems can be configured to send alarms of different levels to a Director.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure:

Configure one NSX to send level 4 and higher alarms to a Director. Leave the default setting in place for a second NSX. Force these systems to generate alarms and verify that alarm notifications of different levels were sent to the Director.

2.3.3 Information Can Be Propagated From One Director To Another

In addition to NSX systems being able to send alarms to multiple Directors, a Director can forward alarms to other Directors.

Test Scenario

Goal:

Verify Director-to-Director communication.

Reference:

Chapter III in the *NetRanger User's Guide*.

Procedure:

Configure a Director to forward alarms from an NSX to a second Director using the **DupDestination** entry in */usr/nr/etc/smid.conf*. Force the NSX to generate alarms and verify their arrival at both Director systems.

3. Director Capabilities

3.1 *Centralized Command and Control of Multiple NSX Sensors*

The NetRanger Director is designed so that a small team of security technicians can use one Director to manage up to 100 NSX 2000 sensors. In order to minimize the time required to master operation of NetRanger, the Director has been integrated with HP's OpenView network management system. This provides the Director with an industry standard command and control interface. Alarms icons and underlying security information is presented via standard OpenView displays while all system configuration is managed via Java-based interface add-ons.

3.1.1 Provides "Real-Time" Data Collection and Display from Multiple NSXs

In addition to displaying alarm notifications in OpenView, the Director stores this alarm information in flat files. The time it takes alarms to arrive at a Director is in large part a function of network transmission and VPN overhead. Nevertheless, most alarms arrive within two to three seconds *from the point at which a pattern of misuse has been detected.*

Test Scenario

Goal:

Verify that a Director collects and displays information from multiple NSX systems in *near* real-time.

Reference:

Chapter IV in the *NetRanger User's Guide*.

Procedure:

Verify via the */usr/nr/bin/nrconns* command that the Director has "*[Established]*" connections to more than one NSX. Then direct multiple attacks against several NSX systems. Verify that the attack information has been properly logged in the Director's flat files. Compare the times of the alarms, which are when the NSX detected them, with the time the alarms were entered in the flat files. This will show you the propagation and processing delays for the NetRanger system at that time for the given network.

3.1.2 Provides Remote NSX Management

A core NetRanger feature is its ability to manage remote NSX systems from one or more Director systems. The Director provides two interfaces for this task: a Java-based Graphical User Interface (GUI) called **nrConfigure**, and a command-line interface. nrConfigure is the preferred interface for NSX management. The command-line interface serves as a backup when operators need to reconfigure an NSX system from a location that does not have access to a Director system. Both interfaces are built on top of a SNMP-like token protocol that allows an operator to *get* and *set* parameters. All remote configuration is conducted via secure VPNs.

Test Scenario

Goal:

Verify that a Director can remotely manage multiple NSX sensors.

Reference:

Chapter IV in the *NetRanger User's Guide.*

Procedure:

Begin testing by performing several simple *get* and *getbulk* commands against one or more NSX services. This involves the following steps:

1. Select an NSX **icon** from a Director security map
2. Select **Security_Configure** from the OpenView menu. Wait for the nrConfigure interface to come up.

3. Select **postofficed**, **DestinationConnectionStatus**, and **getbulk** from the *Applications*, *Tokens*, and *Actions* list boxes.
4. Click the **Execute** button

This will return a listing in the *Results* box of all current connections to other systems. The same information can be retrieved via the following command-line entry:

```
nrgetbulk 10000 10 100 1 DestinationConnectionStatus
```

where the values "10000", "10", and "100" represent hypothetical *Application*, *Host*, and *Organization* identifiers that nrConfigure provides for you automatically; the "1" is a delivery priority, which always defaults to this value. Please note that the same results can be obtained by *telneting* into the NSX and running the */usr/nr/bin/nrconns* command locally on that host.[5]

Next use **set**, **unset**, and **exec** commands to change configurations on the remote NSX. A good exercise might be to simulate a DEFCON1 to DEFCON2 status change by creating two different configuration files for **sensord** on a remote NSX. This can be done as follows:

1. Create one */usr/nr/etc/sensord.conf* where alarms are displayed but no action is taken. Then create a second file where all alarms cause the attacking system to be shunned. Name the first file *sensord.conf* and the second *sensord.shun.*[6]
2. Confirm that the NSX configuration is based on the passive *sensord.conf* by bringing up nrConfigure and executing **get** on the token **FilenameOfConfig** for **sensord.** The file name "sensord.conf" should be displayed in the "*Results*" box. Verify that shunning has been suppressed by generating several attacks against hosts protected by that NSX.
3. Reset *sensord's* configuration file to *sensord.shun* by executing **set** on **FilenameOfConfig.** A status of *Success* should be returned in the *Results* box. Please note that *sensord* will continue to work from *sensord.conf* settings until it is instructed to re-read its configuration file. This is accomplished by executing an **exec** against the **ReadFileConfig** token.
4. Verify that shunning is active by repeating the attacks performed in step #2.

As mentioned earlier, all of these steps can also be performed via the command-line interface.

[5] In order to run *nrconns* the */usr/nr/bin/nr.configd* daemon must be running on the NSX. This service is automatically started by */usr/nr/bin/nrstart.sh* if there is a "configd" entry in the */usr/nr/etc/daemons* file.

[6] The simplest way to do this is to *telnet* onto the NSX system and modify the files *in-situ* with a tool such as *vi.*

3.1.3 Allows An Operator To Respond to Events

Although NSX systems can be configured to automatically respond to misuse, responses can also be initiated manually by an operator at a Director console. For example, an operator can instruct an NSX to shun a connection or to begin logging all keystrokes. In many cases (e.g., shunning) this type of manual intervention is preferable over an automated response.

Test Scenario

Goal:

Verify that an operator can manually control an NSX system's responses to events from a Director system.

Reference:

Chapter IV in the *NetRanger User's Guide*.

Procedure:

Highlight an alarm associated with one of the NSX systems displayed on a Director security map. Select **Security_Show Names** from the OpenView menu. The Director will display the names and IP addresses of the source and destination of the attack. Apply and remove the shun for this attack based on the procedure documented in section 1.3.2.

3.2 Director Display is Configurable and Flexible

3.2.1 Provides A Complete Description of Security Events

The NetRanger Director provides a description of each security event that occurs.

Test Scenario

Goal:

Verify that the Director provides a description of each security event.

Reference:

Chapter IV in the *NetRanger User's Guide*.

Procedure:

Highlight an alarm associated with one of the NSX systems displayed on a Director security map. Select **Edit_Describe/Modify** from the OpenView menu or press **Ctrl/O.** Then select **NetRanger** from the ensuing *Attributes* dialog. The system will display the time of the event, event type, alarm level, source address, destination address, and so forth. Please note that this information varies by event.

3.2.2 Allows A User To Define Display Thresholds

The operator can define *normal*, *marginal*, and *critical* display thresholds within the NetRanger Director system. These thresholds are set for each NSX. The default is 0-2 for normal (green) status, 3-4 for marginal (yellow) status, and > 4 for critical (red) status. As stated earlier, the NSX assigns the alarm level based on the local configuration. The display status will propagate upward within the display from the incoming alarm.

Test Scenario

Goal:

Verify that multiple NSX systems can transmit different alarm thresholds.

Reference:

Chapter IV in the *NetRanger User's Guide*.

Procedure:

Force three different NSX systems to generate level 2, 3, and 5 alarms respectively. Verify that the default Director settings produce green, yellow, and red icons. Clear the alarms icons and reset the display to all green. Now reset the marginal threshold for one of the NSX system's *sensord* application to level 2 and the critical threshold to level 6 on another NSX. Run the same attacks and verify the settings produce yellow, yellow, and yellow icons.

3.2.3 Minimizes Screen Clutter Through Alarm Icon Consolidation

Many OpenView applications display a separate icon for every monitored event. This can quickly lead to a cluttering of both the user interface and the underlying database, which in turn can impact usability. This is especially true when a Director has been configured to monitor level "1" alarms, such as ping sweeps.

The Director consolidates duplicate alarms into a single icon and Open-View database entry. A counter on the icon indicates when a subsequent alarm of that type has been received. This significantly reduces the number of alarm icons an operator must keep track of. The consolidation threshold for duplicate alarms is configurable.

Test Scenario

Goal:

Verify that a Director can consolidate duplicate alarm events into a single icon.

Reference:

Chapter IV in the *NetRanger User's Guide*.

Procedure:

The default consolidation threshold for duplicate alarms is "2". Verify that the Director is properly configured to display an alarm for the type of attack you plan to test with, then launch that attack three or four times against a host protected by one of the NSX test systems. After the third attack is detected, the Director should display a single alarm icon with a counter that identifies how many instances of that attack have been detected. The OpenView database is also reset to maintain a single record for the attack.

3.2.4 Hierarchical Propagation of Alarms

Once alarm thresholds have been established, a Director can propagate alarm status upward through the icon hierarchy. This icon hierarchy is based on the following levels: NSX *application*, NSX *system*, NSX *collection* (optional), and finally the *root NetRanger* icon. Propagation of alarm status up through the icon hierarchy allows you to visually detect events that have occurred at a level of the hierarchy other than the one currently being displayed. This capability is based on standard OpenView functionality.

Test Scenario

Goal:

Verify that a Director can propagate alarms up through an icon hierarchy.

Reference:

Chapter IV in the *NetRanger User's Guide*.

Procedure:

Generate an alarm icon of marginal status and verify that it propagates up through the Director display hierarchy. Reset the display and repeat with an alarm of critical status.

3.2.5 Supports Multiple, Simultaneous Operators in an Operations Center

The NetRanger Director can be configured to display on multiple consoles in a Network Security Operations Center (NSOC). This is important when management of the security perimeter requires more than one operator. Each display, or *instance*, of the network management interface can be configured to be read-only (essentially a passive display) or read-write (an active display tailored to the operator). These capabilities are also based on standard HP OpenView functionality.

Test Scenario

Goal:

Verify that the Director can provide simultaneous displays

Reference:

Chapter IV in the *NetRanger User's Guide.*

Procedure:

Open multiple read-only instances of the network map and set up several read-write instances of the Director. Verify that they conform to standard OpenView functionality.

3.3 Can Notify Off-Duty Personnel Of Events

In addition to displaying and logging events, a Director can be configured to send e-mail, or e-mail-enabled pages, to an operator. This is especially useful during off-duty hours when an operator is "on call." The response/notification is triggered by user-defined thresholds.

Test Scenario

Goal:

Verify that the Director can send pages and e-mail in response to an event.

Reference:

Chapter IV in the *NetRanger User's Guide.*

Procedure:

Configure a Director to generate pages or distribute e-mail based on thresholds defined in */usr/nr/bin/eventd/eventd.conf.* Then verify that e-mail and/or pager notifications are generated in response to test attacks. Compare the output of these notifications with information associated with the corresponding Director alarm notifications.

3.4 Stages Data To A Relational Database for Subsequent Analysis

As the *NetRanger User's Guide* explains, the Director writes incoming alarm information to flat files in order to maximize performance and fault tolerance. While these flat files can be used as the main data repository in a base NetRanger system, their primary purpose is to serve as a temporary data store for marshalling data into a relational database (such as Oracle), or a trouble ticketing system (such as Remedy ARS).

The process that stages data from the flat file onto a database is built around a lightweight scheduler known as **sapd.** This process automatically pushes data into a database based on *size* and *time* thresholds specified in */usr/nr/etc/loggerd.conf.* The *sapd* scheduler is shipped with an application known as **sapx** that understands how to export data to both Oracle and Remedy. *sapx* also maintains error logs and rolls back incomplete data loads when an error is detected. Bulk loader examples for Oracle are also shipped with this package. Customers who use other database products, such as Informix or Sybase, can use these scripts as templates for developing their own bulk loader scripts, which can easily be inserted into *sapd* in place of *sapx.*

A small number of **sapr** SQL queries are also shipped with the *sapd* package. These queries show how NetRanger data can be accessed relative to the three major data *dimensions* implicit in most security data: **space, time,** and **event.** With NetRanger, *space* translates into network and port addresses; *time* translates directly and includes GMT as well as local time stamps; *event* translates to *security* events and encompasses level, signature type, incoming verses outgoing data, and so on. These queries should be used as a starting point for the development of more complete reports via third-party reporting and analysis tools.

Test Scenario

Goal:

1. Verify that a Director can stage event data to a relational database.
2. Verify that a Director detects data staging errors and rolls back incomplete data transfers.

Reference:

NetRanger User's Guide, Chapter V and Appendix B.

Procedure 1:

1. Install Oracle and configure access to it (Appendix B).
2. Configure SAPD (Chapter V).
   - set DBUser and DBPass in /usr/nr/etc/sapd.conf
   - add nr.sapd to /usr/nr/etc/daemons
1. Verify that log files are being created.
   - */usr/nr/var* should contain the current log file. */usr/nr/var/new* should contain one or more files queued for database transfer. The number of files is a function of alarm activity as well as the */usr/nr/etc/loggerd.conf* time/size thresholds.
   - If there are only a few log files in */usr/nr/var/new*, decrease the serialization control tokens for loggerd so that several new files are created without having to wait a long time:

```
set NumberOfSwitchBytes      = 10000 (10K files)
set NumberOfSwitchMinutes    =       5 (switch every 5 minutes)
```

   - If there are "too many" log files in */usr/nr/var/new*, you can manually remove some of earlier files from */usr/nr/var/new.*
1. Start **sapd.**
1. Generate log events by initiating several attacks against hosts protected by one of your NSX systems.
2. Verify **sapd** run history by executing a **nrget** against the **RunHist** token for **sapd.** The *Result* should look something like the following output:

```
3 : PRE (ok) RUN (ok) POST (ok) (11/20 12:09 , 11/20 12:11 ) log.199611201206
```

This means that the PRE-RUN-POST cycle ran successfully 3 times. You can also check */usr/nr/var/messages.sapd* for native database dbstage messages.

1. Verify that the data has been successfully loaded into the database by running one or more of the **sapr** queries shipped with the Director SAP package. These queries are designed to run with Oracle's **sqlplus** command line query tool. For example, the number of alarms received, by alarm level, can be obtained by executing the **event1** query as follows:

```
sqlplus <connect string> @/usr/nr/bin/sap/sql/event1
```

1. Repeat steps 6 & 7 several times as new log files are generated. Please note that sapd loads older log files first. New alarms do not appear in the database until the previous log file has been loaded.

Procedure 2:

Run this test procedure to verify that *sapd* properly detects data staging errors and rolls back data associated with the current log file. The larger the log file, the easier it will be to validate this claim.

1. Disrupt database connectivity *in the middle* of a *sapx* data transfer. The easiest way to do this is to disconnect the Director host from the network.
2. Assess the state of the data transfer by executing **nrget** against the **RunHist** token. The *Result* should look as follows:

```
2 : PRE (ok) RUN (err) UNDO (ok) (11/20 12:14 , 11/20 12:15) log.199611201231
```

Event to IP Address Matrix
for
NetRanger® Evaluation

Event Number	Action	IP Address
1	Deploy as Bridge or Router	Proven by Static Network Configuration
2	Broadcast to Multiple Locations	Proven by Static Network Configuration
3	Alarms are Configurable	Proven by Static Network Configuration
4	Information Propagated between Directors	Proven by Static Network Configuration
5	Real-Time Collection and Display from Multiple Directors	Proven by Static Network Configuration
6	Provide Centralized Support	Proven by Static Network Configuration (SPOCK Location)
7	Run Level 5 Attack	128.190.161.18 204.37.10.99
8	Change Alarm Level	128.190.161.18 204.37.10.99
9	Verify NSX Configuration	144.51.132.98 205.130.144.22
10	Intentionally Misconfigure and Show Error Logged	144.51.132.98 205.130.144.22
11	Confirm Specific Action is Allowed	128.190.161.18 144.51.132.98
12	Show Failure to Execute Specific Action	128.190.161.18 144.51.132.98
13	Disrupt Connection Between NSX and Director, and Send Attack to NSX	198.154.15.2 204.37.10.99
14	Reconnect and Show Queued Alarm Messages	198.154.15.2 204.37.10.99
15	Verify Comms Path, and Disconnect	144.51.132.98

16	Verify Secondary Comms Path	144.51.132.98	
17	Perform *getbulk* Command to Multiple NSXs from Single Director	144.51.132.98 204.37.10.99	
18	Alter Configuration of Multiple NSXs from Single Director	144.51.132.98 204.37.10.99	
19	Operator applies *shun* to ongoing attack	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22	
20	Attack 3 NSXs to produce Red, Yellow and Green Icons	144.51.132.98 147.51.26.62 204.37.10.99	
21	Reset Thresholds to Produce all Yellow Icons when attack is sent second time	144.51.132.98 147.51.26.62 204.37.10.99	
22	Run Attack Signature 5 or 6 times to all NSXs PORT SWEEP	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22	~~RANGE~~ COULD BE
23	Operators see multiple attacks have consolidated into Single Icon 2 5 (BY DEFAULT)	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22	
24	Run Level 2 Attack Signature ~~PROTOCOL~~ PORT SWEEP W/ STRIDE	198.154.15.2 ✓ 128.190.161.18 ✓ 144.51.132.98 ✓ 147.51.26.62 ✓ 204.37.10.99 ✓ 205.130.144.22 ✓	RANGE
25	Run Level 5 Attack and view propagation	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62	RANGE

		204.37.10.99 205.130.144.22
26	Configure Director to generate automatic Email and/or Pages	144.51.132.98 204.37.10.99
27	Run attack and observe Email and/or Pages	144.51.132.98 204.37.10.99
28	Attempt *rlogin*	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
29	Execute *sendmail* attack	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
30	Attempt to send email with content violation	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
31	Generate *atomic alarm* by telnet into target host	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
32	Run port sweep against host and observe composite alarm	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
33	Run SATAN in heavy mode and observe alarm	198.154.15.2 RANGE 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22

34	Run SATAN in Lite mode and observe alarm	198.154.15.2 RANGE 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
35	Set NSX to autoshun telnet attempt, and observe autoshun	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
36	Attempt telnet and have operator manually respond with shun	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
37	Attempt telnet through sniffer and observe that shun occurs within 3 seconds	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
38	Configure NSX to log keystrokes from specified attack type	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
39	Run attack and verify that keystroke capture occurs.	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
40	Configure NSX to dum Event and IP logs to an archive device	144.51.132.98 204.37.10.99
41	Verify that data has been transferred to an off-line device	144.51.132.98 204.37.10.99
42	Run attack to generate alarm	198.154.15.2 128.190.161.18

		144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
43	Highlight alarm and note information displayed	198.154.15.2 128.190.161.18 144.51.132.98 147.51.26.62 204.37.10.99 205.130.144.22
44	Verify that data has been staged to an relational database	198.154.15.2
45	verify that a director can detect staging errors	198.154.15.2

April 30, 1997

SPOCK Netranger Demonstration Group

Dear Participants:

Attached is the DRAFT of a sample letter format for your activity to prepare and forward to NSA.

The NSA General Council has reviewed our proposed Proof of Concept demonstration materials and processes. They created this DRAFT with the essential details they believe are necessary for a full understanding by your management of their concurrence and what that agreement encompasses.

Please forward it to your management with a background on the demonstration details.
Refer to all planning documents supplied during the 14 February planning exercise at COACT.

We would like a response, if possible, by Tuesday, 25 February, 1997. Provide a signed copy of your site's demostration approval for 'Terrance M. Losonky, SPOCK Program Manager, National Security Agency, Office of Commercial Solutions and Enabling Technologies,' authorized by the level your command deems appropriate.

Forward it to CPT Artiaga, NSA/V2 c/o COACT Inc. Their FAX is 301-498-0855. Questions, phone (301)498-0150 or (410)859-6318.

Sincerely,

J.E. Artiaga
CPT, AD
NSA/V2
SPOCK Coordinator

FOR OFFICIAL USE ONLY

SECURITY PROOF OF CONCEPT KEYSTONE

NETRANGER REAL-TIME NETWORK INTRUSION DETECTION

Performance and Security Test

Appendix B

Participant Verification and Comments

Prepared for:
Maryland Procurement Office
9800 Savage Road
Fort George G. Meade, Md. 20755

Contract:
MDA904-96-C-0215

By
COACT, Inc.
9140 Guilford Road, Suite L
Columbia, Maryland 21046

Document No. 010511

30 April 1997

Not releasable to the Defense Technical Information Center Per DoD Instruction 3200.12 Distribution limited to U. S. Government Agencies Only.
This document contains NSA Information (Jan.1992). Request for this document must be referred to the Director NSA.

FOR OFFICIAL USE ONLY

Appendix B

Introduction

This Appendix contains information related to the SPOCK (Security Proof of Concept Keystone) report on the WheelGroup NetRanger Real-Time Network Intrusion Detection Product.

Content. The following materials are presented in this Appendix:

1) Master Interrogation Form,

2) Instruction for Completing the Interrogation Form,

3) Participant Comments.

Master Interrogation Form

The Master Interrogation Form was the template used during the test exercises to record the test results. The form contains the following sections:

1) Reference Event Number (event log reference number),

2) Script Reference (data files used during exercises for input/penetration/exercising, etc.),

3) Date/Time (when tests conducted),

4) Location (where tests orchestrated from),

5) Participants (who was present during tests),

6) Hardware Complement (model/designations of major equipment used during tests),

7) Software Complement (generic names of major software used during tests),

8) Test Equipment (mode/designations of major test equipment used during tests),

9) Major Events (plain English dialog of events),

10) Result Summary (observer conclusion).

Instructions for Completing the Interrogation Form

The instructions for completing the Interrogation Form were distributed to all participants in the testing. These instructions explain how to use the master form, the test scripts and the Event Log. For more information regarding the test scripts and the Event Log, see Appendix A: Scripts Used During Testing.

Participant Comments

The Interrogation Forms containing the comments of the participants are included in this appendix. If participants submitted handwritten forms, then those forms were accurately transcribed verbatim as written. The comments of the participants clearly indicated that each claim was successfully verified per the agreed arrangements (i.e. participating sites for each referenced event, specific claims being tested by each event, etc.). Furthermore, during testing telephone banks were organized between all participants such that all claims could be verified verbally prior to proceeding to the next test. Additional comments that extended beyond the claims being tested were also provided by some participants on the Interrogation Forms.

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Script Reference: (data files used during exercises for input/penetration/exercising, etc. Supply references on disk if possible for inclusion in final report, AND HARDCOPY FOR SURE.)

Date/Time: (when tests conducted)

Location: (where tests orchestrated from)

Participants: (who was present during tests)

Hardware Complement: (model/designations of major equipment used during tests)

Software Complement: (generic names of major software used during tests)

Test Equipment: (model/designations of major test equipment used during tests)

Major Events: (plain English dialog of events)

Result Summary: (observer conclusions)

INSTRUCTIONS FOR COMPLETING FORMS

1. Complete ONE form with names of participants, location, etc. for reference.

2. Reproduce that form. You will need at least 45 (i.e. one for each event)

3. 'Reference Event Number' block on each form should be completed. The final result should be a 'Blank" Master form, One Master form filled in with names, location, and other pertinent facts, and that form reproduced 45 times. Each of those reproduced forms is then completed in advance of the test with the *event numbers entered.*

These forms are used to <u>verify the completion of each 'event'</u> as noted on the Event log (spreadsheet).

4. As the test begins, use the *spreadsheet* (included in this packet). The *event numbers* are at the bottom. The title of each *sequence* is along the top of the column, directly above each event number. The *claims* being verified are shown on the left of the spreadsheet for reference.

6. The detailed description of each event is contained in the document titled *NetRanger Capabilities and Test Scenarios, Version 1.2, dated 24 February 1997*. This document is <u>indexed by paragraph</u>. Note that the event log spreadsheet has <u>corresponding paragraph numbers in parentheses next to each claim</u> (see left column.)

These scripts should be read and understood in advance. Any questions may be directed to Larry Phillips, BTG, and other members of the government team, which will be at COACT, setting up the hardware, connectivity, phones, etc. 301-498-0150, or fax 301-498-0855.

7. During the test, a *conference phone*, plus fax and possibly E-mail, will be used to keep everyone 'on schedule'.

8 **After the test, mail the forms to:**

COACT Inc.
Attn: CPT Jay Artiaga
9140 Guilford Road, Suite L
Columbia, Md. 21046

9. Finally, we have worked this as a group, *done our homework*, and although somewhat adventurous, we can stop if necessary, breathe, and straighten anything out. That is the strength of SPOCK's way of doing business. Good people, solid goals, and the desire to succeed!

AFIWC REPORTS

27 Mar 97

MEMORANDUM FOR CAPT ARTIAGA, NSA/V2 COACT

FROM: AFIWC/EASL (1LT IBAÑEZ DSN 969-4760)
250 Hall Blvd, Ste 139
San Antonio, TX 78243,7063

SUBJECT: NetRanger WAN Performance Evaluation

The NetRanger Wide Area Network performance evaluation was conducted during the week of 17 March - 21 March 1997. The evaluation was performed in concert with 7 geographically separated test sites. A total of 45 events were tested among the test sites. The AFIWC/EASL Technology and Network Security Test Facility participated in the evaluation of 23 events. The following summarizes the test results of the 23 events observed and/or performed by the AFIWC/EASL Test Facility.

Note: The events will be numbered according to the Evaluation Events Spreadsheet provided to all the test sites.

Event 13 and 14
Disrupt connection between Director and NSX temporarily and reestablish connection to observe the alarms generated by the SPOCK test site. This event tests the NSX's ability to hold alarms while connectivity with the Director is disputed.

Results: The NSX successfully sent the queued alarms to the Director when connectivity was restored.

Event 19
Apply manual shun to ongoing attack. The SPOCK test site launched an attack in order to generate an alarm. This event tests the NetRanger's ability to allow the operator manual shunning capability from the Director.

Results: The manual shun was successfully executed using the Director's nrConfigure GUI tool.

Event 22 and 23
A series of attacks was launched from the SPOCK test site. This event tests the NetRanger's capability to consolidate multiple alarms from the same source into a single icon.

Results: The Director generated 42 icon-alarms - 3 red and 39 yellow. The red alarms appeared 3 minutes apart from each other and were not consolidated into one icon. The yellow alarms had more than one "lightening bolt", symbolizing multiple alarm consolidation into a single icon.

Event 24 and 25
A level 2 attack was launched. A level 5 attack was launched a couple of minutes after the initial level 2 attack. The purpose of this test is to verify alarm status propagation by the Director. The propagation follows the icon hierarchy outlined in section 3.2.4 of the Capabilities and Test Scenarios document.

Results: The Director successfully displayed both level 2 and level 5 alarms. The icon hierarchy was properly displayed on the Director console.

Event 28
Attempt to log into a protected machine via *rlogin* service. This event verifies NetRanger's default configuration to disallow*rlogin* requests and trigger a level 2 alarm.

Results: NetRanger successfully detected, disallowed, and alarmed the *rlogin* request.

Event 29
A *sendmail* attack was launched from the SPOCK test site. This event verifies the NetRanger's ability to detect a typical*sendmail* attack.

Results: NetRanger successfully detected and alarmed on the *sendmail* attack.

Event 30
An e-mail message with "dirty-strings" (as defined by NetRanger's security policy) was sent. This event verifies the NetRanger's ability to detect dirty-strings passing through the sendmail port.

Results: The dirty-string was successfully detected by NetRanger and a visual alarm was sent to the Director.

Event 31
Telnet to protected target and trigger an atomic alarm by entering a dirty-string. This event verifies NetRanger's ability to detect dirty strings passing through the*telnet* port.

Results: NetRanger successfully detected the dirty strings and sent a visual alarm to the Director.

Event 32
A port sweep attack was launched by the SPOCK test site. This event tests NetRanger's capability to detect and alarm on "composite" attacks.

Results: NetRanger successfully detected the attack and sent a visual alarm to the Director.

Event 33 and 34
These events tested NetRanger's ability to detect normal SATAN attacks and heavy SATAN attacks.

Results: The AFIWC/ EASL Test Facility did not participate in these events.

Event 35
This event tested NetRanger's ability to autoshun an offending host during a telnet attempt.

Results: Due to network connectivity problems the AFIWC/EASL Test Facility did not participate in this event.

Event 36
This event tested NetRanger's ability to allow the operator to manually shun an offending host during a *telnet* session.

Results: The offending host was successfully shunned using the Director's nrConfigure tool.

Event 37
During a monitored *telnet* session, the offending host was to type dirty strings to trigger a shun by NetRanger. The purpose of this event was to validate NetRanger's capability to shun within 3 seconds.

Results: The AFIWC/EASL Test Facility did not participate in this event.

Event 38 and 39
The purpose of this test is to verify NetRanger's ability to log a session -keystroke by keystroke- from beginning to end.

Results: NetRanger was successfully configured to log an entire session. The session was captured in raw-data format.

NOTE: The ability to transcribe the session into a human readable format was not tested.

Event 42 and 43
The purpose of these events was to generate a visual alarm and have the Director display pertinent information on that particular alarm (i.e. source, destination, time, date, etc.).

Results: The alarm was successfully displayed on the Director's console and all pertinent information on that alarm was able to be displayed as well.

Event 44
This test validates NetRanger's ability to stage all captured data onto a relational database for further retrieval and analysis IAW paragraph 3.4 of the Capabilities and Test Scenarios document.

Results: All data was successfully staged to an Oracle data base server. Several queries were performed in an attempt to analyze detection activity during a predetermined period of time (see Attachment 1).

Event 45
This test validates the Director's ability to detect data staging errors and roll back an incomplete transfer IAW procedure 2 of paragraph 3.4 of the Capabilities and Test Scenarios document.

Results: The database connectivity between the Director and the Oracle server was disrupted in order to trigger a data staging error. The staging error was successfully verified by checking the state of the data transfer.

MARIO A. IBAÑEZ, 1Lt,
USAF
Test Facility Administrator

Attachments:
. Event 44 - Query Results

Event 44 - Validation

Query Results
This attachment shows the actual query session from the Oracle client loaded in the Director to the Oracle database server.

```
netrangr@omega:/usr/nr/bin/sap/sql
>sqlplus netranger/wgcnr@NRDEMO

SQL*Plus: Release 3.3.2.0.0 - Production on Thu Mar 20 16:35:45 1997

Copyright (c) Oracle Corporation 1979, 1994. All rights reserved.

Error accessing package DBMS_APPLICATION_INFO
You may need to install the Oracle Procedural option
SET APPINFO requires Oracle Server Release 7.2 or later

Connected to:
Oracle7 Server Release 7.1.3.0.0 - Production Release
With the distributed and parallel query options
PL/SQL Release 2.1.3.0.0 - Production

SQL> @event1
event1.sql
This query counts how many times each NSX signature has been
detected. Signatures counts are presented in descending order.
The amount of data that is returned to you is constrained by
by a START_DATE and a MIN_EVENT_COUNT.

1: Enter start date (required)
2: Enter a minimum count threshold (required)
```

```
1: START_DATE (MM/DD)> 03/20
2: MIN_EVENT_COUNT> 1
please wait...query in progress

SIGNATURE                   COUNT
------------------------ ----------
Sec Violation                 367
String Match                   20
TCP Port Sweep                  6
ICMP Src Quench                 4
Mail: recon                     3

SQL> @event3
event3.sql
This query counts the number of unique NSX string alarms. It
can also be used to return a count for a specific string alarm.
The amount of data that is returned to you is constrained by a
Atch 1
START_DATE, a MIN_EVENT_COUNT and an optional MATCH_STRING.


1: Enter start date (required)
2: Enter minimum count threshold (required)
3: Enter alarm string (optional)

1: START_DATE (MM/DD)> [illegible]
2: MIN_EVENT_COUNT> 1
3: MATCH_STRING>
please wait...query in progress

SUBSTR(DATA_ALARM,1,[illegible]                        COUNT
--------------------------------------          ------
INCOM1_TCP_FAIL                                    355
IFS=/                                               14
/etc/shadow                                          5
INCOM1_FINGER_FAIL                                   5
INCOM1_DNS_FAIL                                      4
vrfy                                                 3
INCOM1_FTP_FAIL                                      2
INCOM1_LOGIN_FAIL                                    1
foobar                                               1

9 rows selected.

SQL> @event4
event4.sql
This query counts NSX str[illegible]                  [illegible]ate type. It
can also be used to return a [illegible]              [illegible] string alarm.
The amount of data that is ret[illegible]             [illegible]nstrained by a
START_DATE, a MIN_EV[illegible]                       [illegible] an optional MATCH_STRING.

1: Enter start date (required)
2: Enter minimum count thresh[illegible]
3: Enter alarm string (optional

1: START_DATE (MM/DD)> [illegible]
```

```
2: MIN_EVENT_COUNT> 1
3: MATCH_STRING>
please wait...query in progress
```

Atch 1

```
STRING                                          SIGNAME                 COUNT
----------------------------------------------- -------------------- --------
INCOM1_TCP_FAIL                                 Sec Violation             355
IFS=/                                           String Match               14
/etc/shadow                                     String Match                5
INCOM1_FINGER_FAIL                              Sec Violation               5
INCOM1_DNS_FAIL                                 Sec Violation               4
vrfy                                            Mail: recon                 3
INCOM1_FTP_FAIL                                 Sec Violation               2
INCOM1_LOGIN_FAIL                               Sec Violation               1
foobar                                          String Match                1

9 rows selected.

SQL>
```

Atch 1

BCBL REPORTS

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 19

Date/Time: (when tests conducted)
19 March 1997
20 March 1997

Location: (where tests orchestrated from)
BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)
Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG - Borderguard 2000

Major Events: (plain English dialog of events)

1444 Received TCP Port Sweep Alarm
Source IP - 208.213.191.4

Manually Shunned Source IP
Was informed that shunned IP Address was unable to TELNET into our Network.

1450 Unshunned Source IP

Result Summary: (observer conclusions)

Test was successful.

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 20

Date/Time: (when tests conducted)
19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)

Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

Reset critical alarm to Level 3

0910 Received Mail Recon Alarm
Severity 3 Alarm

ICON color was red instead of yellow

Result Summary: (observer conclusions)

Operation was able to change color or alarm severities

Test was successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 23

Date/Time: (when tests conducted)

19 March 1997
20 March 1997
Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R. J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)
Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

Instead of multiple alarms for an alarm condition that is identical to a previous alarm condition (with identical parameters - Source IP, Target IP) - only one alarm ICON was generated.
The number of attacks tallied on the one ICON

Result Summary: (observer conclusions)

Test was successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 28

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R. J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)
Target - Sun Sparc 5
Director - Sun Ultra
NetRanger- NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

1240 Received LOGIN FAIL alarm
Severity level 3
Source IP 208.213.191.4

Result Summary: (observer conclusions)
Alarm was observed

Test was successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 29

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)

Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSGBorderguard 2000

Major Events: (plain English dialog of events)

1246 Received Mail:Recon alarm
Severity 3
Source IP 208.213.191.4

Result Summary: (observer conclusions)

Alarm was observed
Test was successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 31

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)

Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

1305 Received MATCH: IFS=/ alarm
(2 - one in, one out)
Severity 5
Source IP (in) 208.213.191.4

Result Summary: (observer conclusions)

Alarm was observed

Test was successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 32

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model designations of major equipment used during tests)

Target - Sun Sparc 5
Director - Sun Ultra
NetRangeer - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

1343 Received
one TCP Port Sweep alarm
Severity 5
IP Source 208 213 191.4

10 Incom1-TCP_Fail alarms
Severity 3
IP Source 208 213 191.4

Result Summary: (observer conclusions)
Alarm observed
Test was successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 35

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)
Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

1558 Received
Two MATCH: IFS=/ alarms
(one in, one out)
Severity 5
IP Source (in) 208.213.191.4
(out) 147.51.26.62
Verified NrManaged
Shun Host List
208.213.191.4
147.51.26.62

Result Summary: (observer conclusions)
Verified shunned IP addresses
Test successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 36

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)

Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

1508 One alarm received

MATCH: /etc/shadow
Severity 4
Source IP 208.213.191.4

Operation manually shunned 208.213.191.4

Result Summary: (observer conclusions)

Verified Source IP was shunned
Test successful

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 38 and 39

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model designations of major equipment used during tests)

Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

1552 Received to MATCH IFS alarms
Severity 5
Source IP (in) 208 213 191 4
(out) 147 51 26 62
Checked netranger@bcbl nsx :usr/nr/ver/iplog
and verified two files
iplog. 147.51 26 62
iplog. 208.213 191 4

Result Summary: (observer conclusions)
System did perform keystroke capture his
However difficulty has been noted in translating the files into readable form

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 42 and 43

Date/Time: (when tests conducted)

19 March 1997
20 March 1997

Location: (where tests orchestrated from)

BCBL(G)
Fort Gordon

Participants: (who was present during tests)

R.J. Casella (BCBL)
J. Widby (BCBL)
R. Dwire (NSG)

Hardware Complement: (model/designations of major equipment used during tests)

Target - Sun Sparc 5
Director - Sun Ultra
NetRanger - NSX
NSG Borderguard 2000

Major Events: (plain English dialog of events)

Various Alarm conditions were generated from attack IP Source as were an IP Source located locally.

All attacks were noted and caused alarm conditions.

All alarms were highlighted and alarm information was displayed.

Result Summary: (observer conclusions)

Overall successful

COACT REPORTS

Conference Report
(spocmi46)

Minutes of WheelGroup NetRanger Proof of Concept Execution. 24 March, 1997
Date of Conversation: 19-21 March 1997
Personnel in Conversation: See attachment.
Reference: NetRang4.xls file, 'Event Log'.

OLD BUSINESS:

This meeting was held to discuss the execution of the WheelGroup Proof of Concept , complete the hardware connectivity, and assign tasks during its orchestration.
The meeting was opened at 8.00a.m. by Larry B. McGinness, Coact.

NEW BUSINESS:

COACT briefed Terry Losonsky, NSA/V2 on the status of the arrangements for the proof of concept. This consisted of the following topics:

Sites brought up and those still being troubleshot.
The phone bank, established to coordinate with the sites, and the programming of the site's numbers.
The procedures and documents to be used to verify the IP addresses during each event.

Current remaining problems were FIWC connectivity with COACT, and loading SUN Solaris OS into the SPARC 5 loaned by NSA/V2 to COACT (the launch terminal for scripts). A proposed solution to the SPARC problem is to have WheelGroup use their LapTop instead of the SPARC 5.

COACT discussed with WheelGroup the capability of the SPARC20 (monitor terminal) to take snapshots of the monitor screen as events unfolded. It was decided to use the audit log instead (for inclusion in the report), with illustrations from the manuals to show typical types of screens. (This decision had to do with reproduction quality in the reports. The manual illustrations were adjusted for resolution and contrast to be suitable for reproducing.)

Terry Losonsky returned to NSA to brief the DDI for permission to execute the proof of concept.

A sign-in sheet was begun for the attendees/participants.

At 0905, FIWC problem was isolated to a firewall filter, needing local authority to remove.

ACTIONS:

Reference WheelGroup Scripts and Master Event Log (with events referenced to scripts):

COACT maintained the Event Recording of time of launch, and monitored conversations over the phone bank.

BTG manned the Phone Bank, coordinating with sites.

WheelGroup launched the attack scripts, using the laptop, connected via. ISDN 64kbyte through the internet, to the sites, and monitored/communicated with the NSX/Directors using the SPARC 20. Also provided technical support when requested over the phone banks.

NSA/V2 monitored the IP addresses before each launch, and maintained the Master set of observation forms, recording key information on each event.

Event 1 - (1300) A potential 'problem' occurred in verifying the claim that NetRanger can be deployed as a bridge or router.
The scripts to verify this claim were based on using ping sweeps (strobe) attack. It was discovered that some sites in 'bridging' mode would not respond to the pings. We determined that NetSentry is configured (actually the normal architecture of Ethernet) so that non-existent targets, i.e. non-connected equipment, don' t respond with a ping. As the NetRanger ping sweep signature is created to only recognize six or more pings as a ping sweep, and ethernet protocol only provides pings for existing connected hosts, only sites with six or more hosts will trigger the signature alarm *in bridging* mode.

As FIWC did have the requisite hosts, and was also configured in bridging mode, the claim was verified using the pre-arranged scripts for event 1. However, alternate signatures, not requiring six pings, were used at the other bridge sites to verify the claim.

Events 2 - 6 (1305) These are verified by virtue of the existence of the established, functioning network. All sites connected and functioning except BCBL, which is having trouble getting to LIWA and NSA The specific symptom is periodic loss of the encrypted sleeves. The BCBL to COACT connection is better, but not perfect. Our analysis attributes their difficulty to a 256kbyte pipe, being shared by BCBL with the rest of the base, while the other players have T1 connectivity.

Event 7 - (1330) verified.

Event 8 - (1332) verified

Event 9 - (1345) verified

Event 10 - (1350) verified.

Event 11 - (1400) verified

Event 12 - (1405) verified

Event 13 - (1415) verified

Event 14 - (1415) verified

Event 15 - bypassed.

Event 16 - bypassed.

Event 17 - (1420) verified

Event 18 - (1430) verified

Event 19 - (1435) verified

Event 20 - (1500) attempted but not verified.
Event 21 - (1505) attempted but not verified.

End of session for Wednesday, 19 March 97.

Beginning of events for Thursday, 20 March 97.

A review of residual problems from Wednesday:

COACT site not able to TELNET to NSA site. The NSA has a sniffer on the line and is able to read the TELNET packets coming in. NSA has tried to TELNET to themselves to no avail. All assume the NSA system administrator has not enabled proper protocol (50 or 60), this being an NSA action.

The BCBL host is still locking up. The problem has been further isolated to the rented SPARC terminal, which apparently has insufficient memory. (WheelGroup notes that the partition on the disk was reviewed, and although not as large as usually used with their product, they assumed it would be sufficient for the test. Apparently, the memory is being eaten up more than anticipated. WheelGroup is freeing up memory by eliminating superfluous HP OpenView functionality, and clearing out some of the audit log.

Also found an IP error (NSX to Director), which has been corrected.

Events 20 and 21 remain unverified. WheelGroup has reviewed the problem and conveyed software changes to COACT site. The tests can now be re-run.

Event 20 -(0905) repeated and verified successfully.

Event 21 - (0915) repeated and verified successfully.

Event 22 - (0935) verified.

Event 23 - (0930) verified.

Event 24 - (0940) verified.

Event 25 - (0942) verified.

Test halted to troubleshoot the COACT Sparc 5 disk drive, needed to load the SOLARIS OS. This terminal will be needed for phase 2 tests to launch SATAN scripts.

(1230) SPARC 5 up and loaded with SATAN as obtained from the Internet. This version requires the user to register on Domain Name Server. BTG System Administrator contacted and COACT site registered under BTG administration. As NetRanger network was brought up so rapidly, no sites are registered on DNS. Hence, SATAN will not operate.

Event 26 - a review of the script indicates this test was actually run on Tuesday as a by-product of other events. E-Mails being sent to COACT by sites for event verification.

Event 27 - this test was also run on Tuesday as a by-product of other events. E-Mails being sent to COACT by sites for event verification.

Event 28 - (1240) verified.

Event 29 - (1243) verified.

Event 31 - (1301) verified.

Event 30 - (1355) verified.

Event 32 - (1340) verified.

Events 33 and 34 - SATAN will not run from COACT site to participants because of the DNS registration. LIWA agrees to run a modified version (MITRE created) from one of their on-site systems to the test system to verify the claim. All icons from SATAN scripts respond except the combined icon acknowledging this is a SATAN type attack. Reason determined to be the fact that MITRE modified it sufficiently that the SATAN signature, resident in the NSX could not recognize it as SATAN.

AFWIC also running SATAN at their site. All function as claimed **(event 33 and 34) and verification accomplished.**

Note:
AFWIC running SATAN at extremely hi-speed, not encountered when input is limited by internet. Nonetheless, all attacks logged and eventually reported out as I/O allows.

Event 35 - (1435) verified by FIWC. BCBL failed due to faulty host and limited connectivity thruput.

Event 36 - (1445) verified by FIWC. BCBL again failed due to host memory and limited connectivity thruput.

Event 37-(1508) verified.

Event 38 - (1530) verified.

Event 39 - (1530) verified

Event 42 - (1540) verified

Event 43 - (1540) verified

Event 38 - (1545) rerun and again verified.

Event 39 - (1550) rerun and again verified.

Event 42 - (1550) rerun and again verified

Event 43 - (1550) verified

End of session for Thursday, 20 March 97..

Beginning of events for Friday, 21 March 97.

Review of residual problems from Thursday:

The Relational Database population feature needs to be verified. The software functions and appropriate commands need to be furnished by WheelGroup to AFWIC. Thursday, AFWIC was given a Point of Contact at WheelGroup to resolve the operation required to verify the claim. AFWIC has opened the database at their site and found it populated with the information resulting from the testing. Claim has been verified.

The Dual Homing claim could not be verified by NSA because no physical telephone lines could be obtained for use by the NSA/V2 lab (because of the unclassified nature of the test and the sensitivity of

NSA's connection of their phones to outside sources.) The test team has worked to come up with an alternate scenario. A router is being used to sub-divide a physical connection into two logical connections. SOLARIS recognizes two logical lines as dual assignment of an IP to two different locations (i.e. error). Each connection is removed from the router until the new logical line is assigned satisfactorily. Then the other lines are re-added. The claim can now be verified using the two logical lines to demonstrate dual-homing capabilities.

Event 15 - (0935) verified.

Event 16 - (0935) verified.

Event 40 - (1015) verified.

Event 41 - (1015) verified.

A review of the three day's events indicates ALL claims verified by executing all events as of 1100.

Each site is contacted and reminded to complete one Master Observation form, listing all participants, verifying equipment and software complements. This will be attached on individual forms, one for each event the site participates in, and the form completed with satisfactorily verified statement as a minimum. Other comments if desired. Complete packages to be mailed to COACT for inclusion in the report.

NSA/V2 observation forms (done by CPT Artiaga at COACT) collected and placed in master documentation envelope for inclusion in report.

ATTENDEES:

Jay Artiaga NSA/V2
Terry Losonsky, SPOCK- PM
Larry Phillips, BTG
Larry B. McGinness, COACT
Jerry Latham, WheelGroup
Mathew McDonald, CNSG/NGP
Paul Kominos, AGIC(I)

FIWC REPORTS

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 1

Date/Time: (when tests conducted)

March 17, 1997

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1.Creator - Director
Borderguard 2000 - Router
Netranger NSX - Sensor
Gateway 2000 - Target
with Red Hat LINUX

Major Events: (plain English dialog of events)

Deploy as Bridge or Router

Result Summary: (observer conclusions)

Satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 2

Date/Time: (when tests conducted)

March 17, 1997

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1.Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
w/ Red Hat LINUX

Major Events: (plain English dialog of events)

Net configuration will demonstrate

Result Summary: (observer conclusions)

Satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 2

Date/Time: (when tests conducted)

March 17, 1997

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1. Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/ Red Hat LINUX

Major Events: (plain English dialog of events)

Alarms will reflect participant's requirements

Result Summary: (observer conclusions)

satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 4

Date/Time: (when tests conducted)

March 17, 1997

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra1.Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000

Major Events: (plain English dialog of events)

Information can propagate between directors

Result Summary: (observer conclusions)

Satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 5

Date/Time: (when tests conducted)

March 17, 1997

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/ Red Hat LINUX

Major Events: (plain English dialog of events)

Real time collection and display from multiple NSXs

Result Summary: (observer conclusions)

Successfully verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 6

Date/Time: (when tests conducted)

March 17, 1997

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
w/Red Hat LINUX

Major Events: (plain English dialog of events)

Provides centralized support

Result Summary: (observer conclusions)

Satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 7

Date/Time: (when tests conducted)

March 19, 1997, 1330

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Software Complement: (generic names of major software used during tests)

SATAN

Major Events: (plain English dialog of events)

Run level 5 attack

Result Summary: (observer conclusions)
Satisfactorily verified
Level 5 attack observed. In this case, TCP Port Sweep

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 8

Date/Time: (when tests conducted)

March 19, 1997 1332

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
Netranger NSX - Sensor
Gateway 2000 - Target
W/Red Hat LINUX

Software Complement: (generic names of major software used during tests)

SATAN

Major Events: (plain English dialog of events)

Reconfigure and run same attack, showing different Alarm level

Result Summary: (observer conclusions)

Satisfactorily verified
Level 5 attack observed. Threshold for Port Sweep changed to level 4. Port Sweep was observed.

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 13 and 14

Date/Time: (when tests conducted)

March 19th 1415

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/Red Hat LINUX

Major Events: (plain English dialog of events)

Event 13 - Disrupt connection between Director & NSX.
Event 14 - Reconnect Director & view queued alarm messages.

Result Summary: (observer conclusions)

Event 13 - Satisfied, verified
Event 14 - Satisfied, verified
Alarms from the FIWC NSX were queued and viewed on the FIWC Director
For this test only, the network connection was disrupted, How would the Director handle a complete power loss?

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 17

Date/Time: (when tests conducted)

March 19th. 1500

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)
Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/Red Hat LINUX

Major Events: (plain English dialog of events)

Event 17 - Perform 'getbulk' command to multiple NSXs from single Director (FIWC Director)

Result Summary: (observer conclusions)

Success verified

'getbulk' command performed on FIWC NSX and SPAWAR NSX.

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 18

Date/Time: (when tests conducted)

March 19, 1997 1500

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)

Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/Red Hat LINUX

Major Events: (plain English dialog of events)

Alter configuration of multiple NSXs from single Director

Result Summary: (observer conclusions)

Satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 19

Date/Time: (when tests conducted)

March 19, 1997 1500

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)
Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/Red Hat LINUX

Major Events: (plain English dialog of events)

Operator applies 'shun' to ongoing attack

Result Summary: (observer conclusions)

Satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 20

Date/Time: (when tests conducted)

March 20, 1997 0905

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)
Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/Red Hat LINUX

Major Events: (plain English dialog of events)

Attack 3 NSXs to provide red, yellow, and green icons

Result Summary: (observer conclusions)

Satisfactorily verified

(MASTER) *locally reproduce as needed*

Interrogation forms Used During the NetRanger Exercises

REFERENCE EVENT NUMBER: (event log reference number.)

Event 21

Date/Time: (when tests conducted)

March 20, 1997 0905

Location: (where tests orchestrated from)

Fleet Information Warfare Center, NAB Little Creek
Norfolk, VA

Participants: (who was present during tests)
Arnold Llamas, FIWC/QVESTECH
Norm Beebe, NSGA Pensacola
Stacy Smathers, NSGA Pensacola
Lt. Nina Kenmore, FIWC

Hardware Complement: (model/designations of major equipment used during tests)
Sun Ultra 1 Creator - Director
Borderguard 2000 - Router
NetRanger NSX - Sensor
Gateway 2000 - Target
W/Red Hat LINUX

Major Events: (plain English dialog of events)

Reset Thresholds to provide all yellow icons when same attack set is run second time

Result Summary: (observer conclusions)

Satisfactorily verified

'mail:recon' icon was a yellow, level 3 alarm.
'Minimum marginal status' changed to level 4. level 3 alarms subsequently did not appear